

В.М.НОСОВ

**ПРАКТИЧЕСКОЕ
ИСПОЛЬЗОВАНИЕ САВ REDUCE
(НА ПРИМЕРАХ ТЕОРЕТИЧЕСКОЙ
МЕХАНИКИ)**

**Под общей редакцией доктора физ.-мат.наук,
профессора В.И.Стражева**

*Учебное пособие
для студентов инженерно-технических специальностей выс-
ших учебных заведений*

**«ТЕХНОПРИНТ»
Минск 2000**

© Электронный учебник, В.М.Носов, 2003

Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельца авторских прав.

Зарегистрирован в Государственном реестре информационных ресурсов Республики Беларусь.

Регистрационное свидетельство № 1180300281.

УДК 531:681.322–181.4(075.8)

ББК 32.973–01 я 73

Н 84

Рецензенты:

заведующий кафедрой технической механики Белорусского государственного университета информатики и радиоэлектроники,
доктор технических наук, профессор **Сурин В.М.**;

заведующий кафедрой “Теоретическая механика и теория механизмов и машин” БАТУ,
доктор физико-математических наук, профессор **Чигарев Ю.В.**

Носов В.М.

Н 84 Практическое использование САВ REDUCE (на примерах теоретической механики): Учеб. пособие.– Мн.: ТЕХНОПРИНТ, 2000. – 192 с.: ил.

ISBN 985–6373–55–2 ТЕХНОПРИНТ

Пособие посвящено вопросам практического использования на персональном компьютере (ПК) аналитических символьных преобразований и численных расчетов. Излагается система аналитических вычислений САВ REDUCE на примере ее использования в курсе теоретической механики. Основной упор сделан на изложении материала, полезного в повседневной практической работе. При этом излагается минимум сведений, с помощью которых возможно решение большинства практических задач. Оно является необходимым дополнением к широко используемому в ВУЗах и имеющемуся в их библиотеках сборнику заданий для курсовых работ под редакцией А.А.Яблонского (вышедшему в 4-х изданиях тиражом свыше 1 млн. экземпляров), из которого взяты все типовые примеры. С универсальной точки зрения пособие также можно рассматривать как практикум по информатике, в котором для удобства пользования приводятся краткие необходимые сведения по САВ REDUCE и по практической работе на ПК. Описано самостоятельное составление программ для решения различных типовых задач вычислительной математики. Пособие предназначено для студентов инженерно-технических специальностей вузов, инженерно-технических работников, а также для массового пользователя.

УДК 531:681.322–181.4(075.8)

ББК 32.973–01 + 22.21 я 73

ISBN 985–6373–55–2 ТЕХНОПРИНТ

© В.М.Носов, 2000

© Электронный учебник, В.М.Носов, 2003

ПРЕДИСЛОВИЕ

Бурное развитие вычислительной математики и ее программной реализации на персональных компьютерах (ПК) привели к созданию принципиально новых систем аналитических вычислений (САВ) и интегрированных пакетов программ.

До настоящего времени компьютеры помогали в решении математических задач лишь привилегированной научной и инженерной элите, занятой сложными и трудоемкими расчетами. Для их проведения пользователь не только должен был освоить работу на самом компьютере, но математически и алгоритмически правильно поставить задачу, изучить использование довольно сложных численных методов, освоить хотя бы один язык программирования, составить программы расчетов, провести весьма трудоемкую и ответственную их отладку и тестирование.

Первым откровением на фоне разрабатываемых многочисленных САВ стал пакет символьных вычислений REDUCE. Он резко уменьшил затраты времени на их освоение и программирование, а также взял на себя груз алгебраических преобразований огромной сложности. В отличие от языков программирования высокого уровня, таких как Фортран, Си или Паскаль, САВ REDUCE может решать большое количество математических задач путем введения команд, без всякого предварительного программирования. С их помощью на ПК в настоящее время легко реализуется интегрирование и дифференцирование символьных выражений, перестановки и перегруппировки членов, приведение подобных членов, подстановки в выражения с последующим их преобразованием.

Отметим самое важное, что решение задач на ПК может быть получено аналитически, то есть в виде формул, состоящих из математических символов. Вследствие этого САВ называют также системами компьютерной алгебры (СКА), системами символьной математики (ССМ) или современными интегрированными пакетами.

Система REDUCE предоставляет более удобные возможности для автоматического и контролируемого упрощения выражений, преобразования символьных матричных выражений, определения новых функций и расширения синтаксиса программ. Следует отметить ее удивительную простоту и фантастические возможности работы с матрицами, что удобно

использовать как в кинематике, так и при решении систем линейных алгебраических уравнений задач статики.

Для перехода от обычного численного определения реакций связей к получению аналитических зависимостей влияния на них различных факторов (вариации нагрузки, геометрических параметров задачи и т.п.) нужно просто не задавать их значения, записав величины нагрузок и тригонометрические функции углов в символьном виде: силы P , F , $\cos\varphi$ или $\sin\varphi$. При этом греческие буквы в обозначениях углов заменяются их идентификаторами, записанными в латинской транскрипции.

Эта особенность REDUCE позволяет проводить исследование, достойное ранее курсовой работы, над любой задачей статики или кинематики, причем его трудоемкость для студента при этом практически не возрастает. Легкость получения нужных функциональных зависимостей в аналитической форме, никак не определяемых ранее, просто поражает. Ведь для их определения ранее нужно было выполнить численное исследование с трудоемкой подготовкой варьируемых данных в численном виде для каждого рассматриваемого положения, а затем мучиться над обработкой большого массива численных результатов [1].

Система REDUCE отличается тем, что удачно сочетает возможности проведения численных и символьных вычислений с простотой и не слишком высокими требованиями к используемой технике. Это делает ее незаменимой для использования в вузах, компьютерный парк которых в основном морально устарел.

В последнее время разработана версия REDUCE 3.6 под Windows. Однако вследствие своей чрезвычайной редкости из-за строго лицензионного распространения, в учебном пособии рассмотрены более широко распространенные версии 3.0 – 3.5 под DOS, реализуемые на *любых* IBM-совместимых ПК.

Под Windows существует достаточное количество современных систем символьной математики, широко распространяемых на дешевых CD-дисках и в полной мере использующих огромные аппаратные возможности современных ПК: DERIVE 4.11, MathCad 2000, MAPLE V R.5, MatLab 5.2, Mathematica 4.

Рассмотрение же версий REDUCE 3.0 – 3.5 под DOS обеспечивает следующие преимущества:

- создание плавного перехода от языков программирования к использованию современных систем символьной математики;
- удобство работы и простота в освоении;
- необыкновенная легкость перехода к символьным преобразованиям от численных расчетов.

Разработанные современные САВ отражают достижения происходящей в настоящее время в мире информационной революции, на несколько порядков увеличивая интеллектуальные возможности любого человека. При их постепенном освоении возникает ощущение, что ваши способности к точным наукам многократно возрастают. Поэтому использование САВ способно совершить революцию в высшем образовании.

В курсе теоретической механики используются практически все основные разделы вычислительной математики: решение систем линейных алгебраических уравнений при определении реакций, интегрирование при нахождении центра тяжести или замены распределенных сил в статике, численное дифференцирование в задачах кинематики, интегрирование дифференциальных уравнений в задачах динамики. Поэтому применение САВ позволяет значительно повысить эффективность учебного процесса.

Реализации этой идеи с использованием ПК и посвящено настоящее учебное пособие. В нем излагается система аналитических вычислений REDUCE. Книга содержит практические сведения, необходимые пользователю ПК для работы с современными интегрированными пакетами. Основной упор сделан на изложении материала, полезного в повседневной практической работе. Систематизация приведенных сведений позволяет использовать пособие в качестве настольного справочника.

Для реализации пользовательского деятельностного подхода в приложении приводятся краткие необходимые сведения по практической работе на ПК. Описано самостоятельное составление программ для решения различных типовых задач вычислительной математики. Все это делает пособие самоучителем по использованию аналитических методов на ПК. Такой пользовательский или деятельностный подход одновременно с синтетичностью изложения материала обеспечивает возможность в течение 2-3-х дней без предварительной компьютерной подготовки неопытному пользователю освоить работу в нужном объеме с описываемым пакетом САВ.

При изучении курса теоретической механики с применением ПК пособие непосредственно встраивается в учебный процесс, так как является необходимым дополнением к широко используемому в вузах и имеющемуся в их библиотеках сборнику заданий для курсовых работ [22], вышедшему в 4-х изданиях тиражом свыше 1 млн. экземпляров. Из него взяты все типовые примеры рассматриваемых задач, аналитические решения которых методами теоретической механики рассмотрены там же и в настоящем пособии не повторяются.

Пособие посвящено вопросам практического применения компьютерных систем численно-аналитических преобразований на любых IBM-совместимых персональных ЭВМ на примерах решения задач по теоретической механике и математике. Оно является продолжением пособия В.М.Носова “Программирование на персональных ЭВМ задач теоретической механики”, где весь материал рассмотрен для использования численных методов расчета на ПК для тех же типовых примеров. Последнее обстоятельство дает возможность организовать комплексное обучение практическому применению численно-аналитических методов на ПК. Это является отличительной особенностью пособия и представляется важным как с практической, так и с дидактической точек зрения.

Книга может быть также использована при организации учебного процесса с применением ПК любой общетехнической и фундаментальной кафедрой. Вместе с этим пособие можно рассматривать и как практикум по использованию аналитических методов расчета на персональных ЭВМ, в котором для удобства пользования приводятся краткие необходимые сведения по используемой системе аналитических вычислений и по практической работе на ПК. С этой точки зрения все приводимые примеры носят обычный искусственный характер.

При таком подходе изучаются только соответствующие главы пособия, любая из которых может рассматриваться независимо от других. Его может использовать любой, желающий овладеть практическими навыками работы с системами аналитических вычислений, современными интегрированными пакетами программ или уже использующий их для своей инженерной деятельности и имеющий выход на любой IBM-совместимый ПК. Все это делает пособие самоучителем работы на ПК для решения основных вычислительных задач с использованием аналитических методов расчета.

Книга не ставит своей целью охватить всю информацию по структуре и средствам описываемой САВ. Она пытается ознакомить читателя с ее наиболее мощными и удобными возможностями, а также на конкретных примерах научить эффективно применять их в своей работе.

Описываемые в пособии готовые программы для целей организации учебного процесса можно получить у автора вместе с необходимыми консультациями на кафедре теоретической механики БГПА (тел. (8-1037517) 2327425).

В заключение автор считает приятным долгом выразить сердечную благодарность рецензентам, чьи критические замечания и пожелания помогли улучшить пособие.

Следует с особой благодарностью отметить роль заведующего лабораторией ЦИС БГУ доцента Позняка Ю.В., предоставившего в распоряжение автора необходимый справочный материал и стоявшего у самых истоков настоящей работы, которому только неблагоприятные обстоятельства, ко всеобщему сожалению, не позволили принять участие в ее выполнении.

В апробации настоящей работы в учебном процессе и проверке разработанных программ и дополнений в рамках УИРС принимали участие в течение ряда лет студенты машиностроительного, транспортных коммуникаций, роботов и робототехнических систем и строительного факультетов БГПА, а также слушатели семинара ФПКП БГПА. Особенно техническую и информационную поддержку при подготовке пособия оказали студенты Коляда В.М., Грук Е.С., Прилепо О.А., Фей Е.Н., Буча В.В. и Белявский В.И., которым автор выражает признательность.

Текст пособия подготовлен автором на машинном носителе. Все программы и файлы данных помещались в оригинал-макет машинным образом и дополнительно тщательно проверялись. Автор также выражает сердечную благодарность Мальдису Я.А. и Романовскому А.О. за техническую поддержку и консультации при выполнении верстки и выводе на печать оригинал-макета.

Автор

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

В пособии ставится задача дать возможность студенту при изучении курса теоретической механики самому выбирать форму общения с ПК из использования готовых программ, до их самостоятельного составления из приведенных готовых блоков. На каждом уровне возможен выбор различных вариантов использования ПК для решения задачи:

- получение численного решения;
- получение решения в символьном виде;
- исследование аналитического решения на ПК путем варьирования условий задачи или различных параметров.

Для удобства пользования в пособии приведены краткие необходимые сведения для практической работы на ПК и по системе аналитических вычислений REDUCE. Для каждого раздела механики все изложение производится с использованием одного или нескольких типовых примеров, чем достигается целостность восприятия.

Все рассматриваемые примеры взяты из двух последних изданий [22] и [23] сборника заданий под ред. проф. А.А.Яблонского, широко и активно используемого в курсе теоретической механики.

Поэтому пособие может быть использовано как для организации учебного процесса с применением ПК при изучении курса теоретической механики, так и в качестве универсального практикума по информатике и САВ REDUCE. В последнем случае на источник получения рассматриваемых примеров обращать внимание не нужно.

Таким образом, с этой точки зрения пособие представляет собой профилирование курса информатики и вычислительной математики для нужд теоретической механики. Оно может служить основой для координации междисциплинарных связей и организации непрерывной подготовки по активному и согласованному использованию вычислительной техники при формировании будущих специалистов.

Ссылки на главы, параграфы, подпараграфы и пункты обозначаются буквой “п” (например, п. 1, п. 1.2, п. 1.2.1, п. 1.2.1.1). Если материал понимается, то на них обращать внимание не надо.

В пособии используется два вида ссылок при описании операторов и программ: на номер в круглых скобках и на номер без скобок.

Ссылка на номер в круглых скобках идентифицирует:

- весь приведенный сегмент программы, если в следующих строках данного программного модуля нет номеров со скобками;
- или только записи, находящиеся в этой же строке, если следующие операторы идентифицированы также номерами с круглыми скобками.

Нумерация с круглыми скобками сквозная в порядке возрастания по всему тексту пособия.

Ссылка на номер без скобок идентифицирует только сам оператор, находящийся в этой строке.

Номера длинных операторов, расположенных в нескольких строках, указываются в первой или в последней строке.

Номера без скобок относятся к описываемым программам и проставлены только для удобства пояснений. В программе на бланке или на экране дисплея операторы не нумеруются.

При описании вариантов и дополнений к основным программам вставляемые операторы должны располагаться в основной программе по порядку возрастания номеров (без скобок), при этом:

- если номер (без скобок) вставляемого оператора *совпадает* с имеющимся в основной программе, то последний *заменяется*;
- если номер *не совпадает* — то оператор *вставляется* между операторами с соответствующими меньшим и большим номерами в основной программе.

Для удобства пользования пособием применяются следующие **шрифтовые выделения**:

- *все, что вводится* пользователем с клавиатуры, *выделяется жирным шрифтом* и печатается *строчными буквами*: $(x+y*x/2+x*z/3)**3$;
- *результаты вычислений и символьных преобразований* печатаются обычными *строчными буквами без выделения* (хотя в большинстве версий REDUCE они будут представлены на экране ПК или в файле с использованием прописных букв) : $(x^3*(27*y^3 + 54*y^2 *z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216$;
- *команды, флаги, операторы и системные функции* REDUCE печатаются прописными латинскими буквами без выделения;
- *команды MS DOS* в тексте печатаются *прописными латинскими буквами и выделяются курсивом* (с наклоном);

- *флажки и переключатели* Norton Commander и Windows печатаются соответственно латинскими или русскими буквами и **выделяются жирным шрифтом**;
- *названия клавиш* печатаются шрифтом Arial и помещаются в угловые скобки: <Enter>;
- *названия панелей, кнопок и опций* Norton Commander и Windows также печатаются шрифтом Arial.

При повторении описываемых сеансов работ в REDUCE следует обращать внимание на автоматическую нумерацию вводимых команд:

- если номер первой команды приводимого фрагмента *начинается с цифры 1*, то **сеанс работы начинается и все впервые используемые в этом фрагменте переменные являются свободными**;
- нумерация *продолжается* – все впервые используемые в этом фрагменте переменные **могут быть связанными** и иметь значения, заданные им в течение текущего сеанса работы. Поэтому для получения приведенных результатов следует *повторить данный сеанс работы с самого начала*;
- если номер шага не указывается – то приводимый фрагмент является локальным: он не зависит от текущего сеанса работы и все впервые используемые в нем переменные **являются свободными**.

Переменные и углы в программах обозначаются в форме их идентификаторов. При этом греческие буквы записываются в их латинской транскрипции, а для двухсимвольных переменных цифра или буквы I, J, K и N выполняют роль индекса (например A1, B2, P1, X0, X1, TN, TK). Этот принцип используется также при описании фрагментов программ.

Во 2-й главе индекс также может обозначаться строчной буквой за обозначением индексированной величины: реакция R в точке E будет обозначаться Re, в точке F — Rf; горизонтальные и вертикальные составляющие реакции в точке A — Xa и Ya.

Значения углов тригонометрических функций при записи уравнений равновесия указаны в градусах.

ОСНОВНЫЕ ОБОЗНАЧЕНИЯ

ДУ	— дифференциальное уравнение,
ДУ2П	— дифференциальное уравнение второго порядка,
ИДФ	— идентификатор файла (его полное имя с указанием пути доступа к нему),
ИФ	— имя файла,
ОС	— операционная система,
ПК	— персональный компьютер,
САВ	— система аналитических вычислений,
СДУ	— система дифференциальных уравнений,
СЛАУ	— система линейных алгебраических уравнений,
ССМ	— система символьной математики,
ССС	— система сходящихся сил,
ТФ	— тип файла.

ЧАСТЬ 1. СИСТЕМА АНАЛИТИЧЕСКИХ ВЫЧИСЛЕНИЙ (CAB) REDUCE

1. БЫСТРЫЙ СТАРТ

Система аналитических вычислений REDUCE предназначена для проведения точных алгебраических вычислений и преобразований любой степени сложности. Она может выполнять разложение и возведение в степень полиномов и рациональных функций в различных формах, раскрывать скобки, выносить общие множители и проводить автоматическое и контролируемое упрощение выражений, удивительно просто работать с различными матричными выражениями, а также выполнять дифференцирование и интегрирование в символьной форме.

Буквы при вводе можно писать строчные и прописные, однако, после проведения вычислений или аналитических преобразований все результаты представляются с использованием строчных букв. Операции с целыми числами осуществляются с произвольной точностью.

Предваряя описание и изложение возможностей CAB REDUCE, покажем на примере нескольких операторов и команд, что это простой и удобный программный продукт.

Наиболее легко получить много полезных результатов и научиться работать с CAB REDUCE можно в интерактивном режиме, называемом также режимом интерпретатора или режимом командной строки.

1.1 ПЕРВЫЙ СЕАНС РАБОТЫ

После загрузки CAB REDUCE, осуществляемой обычно после запуска на выполнение файла `reduce.bat` или `reduce.exe`, система печатает «шапку» типа: `REDUCE 3.3, 15-Jan-88 ...`

где номер версии (3.3) и дата ее создания могут быть другими для конкретной реализации REDUCE на вашем ПК. Затем система приглашает пользователя ввести информацию в командной строке, печатая в ней

1:

Теперь пользователь может ввести после приглашения (1:) правильное «редьюсовское» выражение или команду, ограниченную точкой с запятой, которая обозначает конец выражения, например, возвести число 2 в 2000-ю степень:

$$1: 2^{**}2000; \quad (1.1)$$

После нажатия клавиши <Enter> система вводит выражение, вычисляет его и выдает на экране дисплея результат — строку (строки) вывода или сообщение об ошибочно введенной команде:

```
1148130695274254524232833201177681984022317702088695200477642
7368257662613923703138566594863165062699184459646389874627734
4711896086305533142593135616665318539129989145312280000688779
1482400448714289269900634862447816154636463883639473170260404
6635397090499655816239880894462960562331164953616422197033268
1344168908984458505602379484807914058900934776500429002716706
6258305220081322362812917612678833172065989953964181270217798
5840404215985318325154088943390209192055495778358967203916008
1957216630582755380425583726015528348786419432054508915275783
882625175435528800822842770817965453762184851149029376 (1.2)
```

Тут же выдается следующее приглашение (2:) вводить новую команду и т.д., например:

$$2: (x+y*x/2+x*z/3)^{*}3; \quad (1.3)$$

$$(x^3*(27*y^3 + 54*y^2*z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216 \quad (1.4)$$

Чтобы лучше понять принципы работы системы REDUCE, давайте рассмотрим эти простые примеры.

Поскольку в алгебраических вычислениях нужны точные результаты, то арифметические операции с целыми числами выполняются практически с безграничной точностью (зависящей только от доступной памяти вашего ПК). Это значение (1.2) числа $2^{**}2000$ состоит из 603 цифр и занимает 8 строк на экране дисплея.

Характерной чертой второго примера является то, что во введенном выражении (1.3) все знаки были напечатаны в одной строке, а в результирующем (1.4) — степени переменных подняты на одну строчку вверх.

Отметим также, что система REDUCE имеет дело с переменными и константами. Однако при вычислении выражения (1.3) переменная оста-

лась сама собой, а вычисление было произведено в соответствии с правилами алгебры. В (1.4) приведены подобные члены, а переменные упорядочены специальным образом. Однако и способ раскрытия скобок, и упорядочивание переменных, и формат вывода, и тому подобное, все это контролируется пользователем.

Для осуществления такого контроля имеются две команды **ON** и **OFF**. Каждый режим работы представлен так называемым флагом (переключателем), который может быть либо установлен (поднят) — **ON**, либо отменен (опущен) — **OFF**. В качестве параметров команд **ON** и **OFF** служат имена флагов. Эти описания устанавливают (**ON**) или отменяют (**OFF**) соответствующий режим работы, определяемый указанным флагом (подробнее см. п. 1.5). Например, опуская флаг **ALLFAC**,

3: **off allfac;** (1.5)

мы отказываемся от выноса за скобки общих сомножителей в преобразуемых выражениях. После этого устанавливается новый режим работы и выдается следующее приглашение (4:).

Результаты каждого вычисленного выражения запоминаются системой в переменной **WS** (от *workspace*) и сохраняются до следующего вычисления, в котором этот идентификатор можно использовать. Переменная **WS** называется *рабочим полем* или *рабочей областью выражения*. Так как на 3-м шаге был изменен только режим работы, то для распечатывания предыдущего выражения (1.4) без выноса за скобки общих сомножителей, нужно просто ввести

4: **ws;**

$$(27*x^3*y^3 + 54*x^3*y^2*z + 162*x^3*y^2 + 36*x^3*y*z^2 + 216*x^3*y*z + 324*x^3*y + 8*x^3*z^3 + 72*x^3*z^2 + 216*x^3*z + 216*x^3)/216 \quad (1.6)$$

Выражение (1.4) не изменилось. Оно просто приняло другую форму. Например, для взятия от него производной по **x** нужно ввести

5: **df(ws,x);**

$$(27*x^2*y^3 + 54*x^2*y^2*z + 162*x^2*y^2 + 36*x^2*y*z^2 + 216*x^2*y*z + 324*x^2*y + 8*x^2*z^3 + 72*x^2*z^2 + 216*x^2*z + 216*x^2)/72 \quad (1.7)$$

Установим опять режим выноса за скобки общих сомножителей в преобразуемых выражениях и вычислим интеграл по **x** от выражения (1.6), чему равно теперь значение **WS**:

6: **on allfac; int(ws,x);**

$$(x^3*(27*y^3 + 54*y^2*z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216 \quad (1.8)$$

В каждом слагаемом (1.6) присутствовало x . Поэтому после взятия от него производной по x и вычисления интеграла по x мы должны опять получить исходное выражение. Установка флага ALLFAC должна представить его в форме выноса за скобки общих сомножителей (1.4), что и видим на экране.

Напоминаем, что **все, что должно вводиться вами** в командной строке, **всегда выделяется нами жирным шрифтом**. Как видите, оно имеет достаточно простой вид.

Обычным шрифтом печатаются результаты преобразований и вычисления CAB REDUCE. Они могут иметь очень сложный вид. Но вам не нужно проверять и анализировать промежуточные выкладки. Желательно только продумать систему тестовых проверок (типа приведенной выше).

Другим часто используемым свойством системы является возможность выдачи результатов в форме, совместимой с Фортраном. Такие результаты могут быть использованы для создания программ на Фортране. Это особенно полезно, если рассматривать работу системы, как некоторый способ генерации алгебраических формул, которые должны быть использованы в качестве основы для обширных численных расчетов. Для этого устанавливаем флаг FORT и набираем ws:

8: **on fort; ws;**

$$\text{ans} = (x^{**3}*(27.*y^{**3}+54.*y^{**2}*z+162.*y^{**2}+36.*y*z^{**2}+ . 216.*y*z+324.*y+8.*z^{**3}+72.*z^{**2}+216.*z+216.))/216. \quad (1.9)$$

Имя ANS присваивается выражению по умолчанию и все его строки печатаются с седьмой позиции с использованием синтаксиса Фортрана. В качестве символа переноса используется точка, сопровождаемая пробелом. Вернемся к обычному режиму работы системы, опустив флаг:

10: **off fort;** (1.10)

Можно вводить за один раз несколько выражений и команд (шаги 6 и 8), но система *помнит каждую из них* и печатает следующее приглашение соответственно 8: и 10:.. Оператор DISPLAY позволяет распечатать и просмотреть столько предыдущих выражений и команд M, сколько их указать в качестве аргумента, например для M=5:

11: **display 5;** (1.11)

- 6: on allfac;
- 7: int(ws,x);
- 8: on fort;
- 9: ws;
- 10: off fort;

Если же в качестве аргумента выступает любое нечисловое выражение, то изображаются все предыдущие вводы.

Ко всем результатам можно обратиться с помощью порядкового номера команды N, который в этом случае печатается без следующего за ним в приглашении двоеточия. При этом возможны два способа.

- *Для использования входного выражения в новом вычислении* нужно ввести INPUT(N). При этом будет выполняться входное предложение, соответствующее указанному номеру (происходит его повторное преобразование, вычисление или отработка соответствующей команды, оператора или описания).
- *Для использования только результатов вычислений* нужно ввести WS(N). При этом *не происходит* повторная отработка соответствующей команды, оператора или описания. В этом случае, если порядковому номеру команды N соответствует описание изменения режимов работы **ON** или **OFF**, то они не будут выполнены. В таком случае не возникает и изменения состояния того флага, которое пользователь намеревался произвести, но *система не сообщает об этом*.

С помощью INPUT(N) и WS(N) можно использовать предыдущие входные предложения и результаты вычислений в любых выражениях, например, INPUT(2)**2/WS(5)**3.

Однако удобнее для возможности произвольного обращения к предыдущим выражениям и результатам использовать оператор присваивания. Этот оператор заменяет значение переменной в левой части оператора на значение выражения, стоящего в его правой части, и имеет в простом случае вид:

$$\text{переменная} := \text{выражение} \tag{1.12}$$

Между оператором присваивания и похожим на него алгебраическим равенством существует очень важное различие. Оператор X:=Y+Z означает для машины: взять содержимое ячейки памяти в символьном или числовом виде, соответствующее переменной Y, сложить с содержимым

ячейки переменной Z и передать результат также в символьном или числовом виде в ячейку памяти, соответствующую переменной X.

Знак “:=” в операторе присваивания имеет значение “необходимо заменить на ...”, а в математике знак равенства означает “то же самое значение, что и ...”.

С помощью операторов присваивания сеанс работы 2 — 10 повторен ниже под порядковыми номерами 12 — 20, а совпадающие результаты преобразований, отличающиеся только наличием идентификатора переменной, имеют тот же номер, но со штрихом.

$$12: \mathbf{tt} := (\mathbf{x} + \mathbf{y} * \mathbf{x} / 2 + \mathbf{x} * \mathbf{z} / 3) ** 3; \quad (1.3')$$

$$\mathbf{tt} := (\mathbf{x}^3 * (27 * \mathbf{y}^3 + 54 * \mathbf{y}^2 * \mathbf{z} + 162 * \mathbf{y}^2 + 36 * \mathbf{y} * \mathbf{z}^2 + 216 * \mathbf{y} * \mathbf{z} + 324 * \mathbf{y} + 8 * \mathbf{z}^3 + 72 * \mathbf{z}^2 + 216 * \mathbf{z} + 216)) / 216 \quad (1.4')$$

$$13: \mathbf{off \ allfac}; \quad (1.5')$$

$$14: \mathbf{tt};$$

$$(27 * \mathbf{x}^3 * \mathbf{y}^3 + 54 * \mathbf{x}^3 * \mathbf{y}^2 * \mathbf{z} + 162 * \mathbf{x}^3 * \mathbf{y}^2 + 36 * \mathbf{x}^3 * \mathbf{y} * \mathbf{z}^2 + 216 * \mathbf{x}^3 * \mathbf{y} * \mathbf{z} + 324 * \mathbf{x}^3 * \mathbf{y} + 8 * \mathbf{x}^3 * \mathbf{z}^3 + 72 * \mathbf{x}^3 * \mathbf{z}^2 + 216 * \mathbf{x}^3 * \mathbf{z} + 216 * \mathbf{x}^3) / 216 \quad (1.6')$$

$$15: \mathbf{ttd} := \mathbf{df}(\mathbf{tt}, \mathbf{x});$$

$$\mathbf{ttd} := (27 * \mathbf{x}^2 * \mathbf{y}^3 + 54 * \mathbf{x}^2 * \mathbf{y}^2 * \mathbf{z} + 162 * \mathbf{x}^2 * \mathbf{y}^2 + 36 * \mathbf{x}^2 * \mathbf{y} * \mathbf{z}^2 + 216 * \mathbf{x}^2 * \mathbf{y} * \mathbf{z} + 324 * \mathbf{x}^2 * \mathbf{y} + 8 * \mathbf{x}^2 * \mathbf{z}^3 + 72 * \mathbf{x}^2 * \mathbf{z}^2 + 216 * \mathbf{x}^2 * \mathbf{z} + 216 * \mathbf{x}^2) / 72 \quad (1.7')$$

$$16: \mathbf{on \ allfac}; \mathbf{tti} := \mathbf{int}(\mathbf{ttd}, \mathbf{x});$$

$$\mathbf{tti} := (\mathbf{x}^3 * (27 * \mathbf{y}^3 + 54 * \mathbf{y}^2 * \mathbf{z} + 162 * \mathbf{y}^2 + 36 * \mathbf{y} * \mathbf{z}^2 + 216 * \mathbf{y} * \mathbf{z} + 324 * \mathbf{y} + 8 * \mathbf{z}^3 + 72 * \mathbf{z}^2 + 216 * \mathbf{z} + 216)) / 216 \quad (1.8')$$

$$18: \mathbf{on \ fort}; \mathbf{tti};$$

$$\mathbf{ans} = (\mathbf{x} ** 3 * (27 * \mathbf{y} ** 3 + 54 * \mathbf{y} ** 2 * \mathbf{z} + 162 * \mathbf{y} ** 2 + 36 * \mathbf{y} * \mathbf{z} ** 2 + 216 * \mathbf{y} * \mathbf{z} + 324 * \mathbf{y} + 8 * \mathbf{z} ** 3 + 72 * \mathbf{z} ** 2 + 216 * \mathbf{z} + 216)) / 216. \quad (1.9')$$

$$20: \mathbf{off \ fort}; \quad (1.10')$$

Обратим Ваше внимание, что после задания переменной ТТ в операторе присваивания (1.3') происходит указание имени переменной при ее распечатке в выражении (1.4'). При изменении же режимов выполнения программы и печати результатов, например при отказе от выноса за скобки общих сомножителей из-за опускания флага ALLFAC в уже заданном выражении (1.4'), происходит только печать перестроенного выражения (1.6') без указания его имени.

Отметим, что значение переменной WS, т.е. последнего вычисленного выражения, может быть присвоено любой переменной при помощи

команды SAVEAS. Следует выдать эту команду с единственным параметром – именем переменной, которой надо присвоить значение переменной WS. Поэтому, чтобы присвоить значение выражения (1.4) из первоначального сеанса работы 2 – 10 переменной TT, нужно после получения выражения (1.4) на 3-м шаге ввести команду

3: **saveas tt;** (1.5' ')

после чего можно сразу начинать повторный сеанс работы 13 – 20, все результаты которого будут совпадать с (1.5')– (1.10').

Оператор присваивания связывает переменную в левой части со значением выражения в правой его части, а команда SAVEAS связывает указанную в ней переменную со значением последнего вычисленного выражения. Эта связь в обоих случаях может быть разорвана командой CLEAR, которая стирает ненужные далее выражения указанных переменных. Например:

21: **clear tt, ttd, tti;** (1.13)

Замечание. Автор ограничен требованиями экономии места, поэтому приводятся довольно простые примеры (кроме 1:), не передающие практически безграничные возможности CAB REDUCE. Попробуйте *сами их проверить*, выполнив операции с более сложными примерами и большими степенями. Возможно, они вызовут у вас удивление. А это очень важное чувство. Когда-то одного маленького мальчика удивила стрелка компаса, показывающего на север. Он стал изучать, думать и размышлять. Вопрос этот однозначно не решен и до сих пор. Но впоследствии фамилия этого мальчика (Эйнштейн) стала известной всему миру. И если при этой проверке благотворное чувство удивления посетит вас и приведет к изучению практического использования CAB на ПК, то ваши возможности также возрастут удивительным образом.

И в надежде на это мы пока завершим наш первый сеанс работы. Команды "BYE; " и "QUIT"; прекращают работу REDUCE и возвращают управление операционной системе. Поэтому

22: **bye;** (1.14)

и приступим к более систематическому изучению.

1.2 ОПИСАНИЕ ЭЛЕМЕНТОВ ЯЗЫКА

1.2.1 Символы и предложения

В качестве основных символов системы REDUCE используются:

- прописные и строчные буквы латинского и русского (там, где он поддерживается) алфавитов;
- десятичные арабские цифры от 0 до 9;
- специальные символы: ! () = - + / * , ; . : % \$ ' " > <

Система допускает при вводе использование в произвольном порядке строчных и прописных букв. Для обычного режима работы CAB REDUCE они внутренне неразличимы. Поэтому для лучшего отличия записи символов всегда будут использоваться:

- в тексте прописные буквы;
- строчными буквами жирным шрифтом с использованием абзацного отступа выделяется все, что должно вводиться в командной строке и после чего должна нажиматься клавиша <Enter> (<Ввод>);
- обычными строчными буквами без использования абзацного отступа печатаются результаты преобразований и вычислений CAB REDUCE.

Предложением в REDUCE называется последовательность символов, заканчивающихся одним из двух знаков “;” или “\$”, называемых разделителями, ограничителями или терминаторами. В первом случае *результат его выполнения* “распечатывается” на экране или в файле (см., например, (1.1), (1.3), (1.4)), во втором — нет (см. (1.17)).

Предложения в программе записываются одно за другим. Их может быть как несколько в одной строке записи, так и каждое из них может занимать несколько строк.

Отметим, что при этом лишние пробелы игнорируются, а знаки переноса не используются.

Каждое предложение, имеющее численное или символьное значение, называется *выражением* (см., например, (1.1), (1.3), (1.4)), не имеющее — *командой* (см. (1.5), (1.10), (1.13)).

Для записи *выражений* в REDUCE используются имена переменных и знаки арифметических операций: + сложение, – вычитание, * умножение, / деление, ** возведение в степень. Например: $B^{**2} - 4*A*C$.

Все знаки арифметических действий должны присутствовать в выражении явно. Математическое выражение IJ (значение I умножается на значение J) в REDUCE должно быть написано обязательно так: I*J, иначе оно будет понято как символическое имя IJ и возникнет трудноуловимая ошибка, не препятствующая работе программы.

Рядом не могут находиться два знака арифметических действий, они должны отделяться скобками, например: $R*(-0.01)$.

Последовательность выполнения арифметических операций в выражении такая же, как и при обычной записи алгебраических формул. Сначала выполняется возведение в степень, затем умножение и деление, а в конце — сложение и вычитание.

Если в выражении записано несколько одинаковых операций подряд, то они выполняются слева направо. В случае необходимости иного порядка выполнения арифметических операций, как и при написании обычных формул, используются круглые скобки.

Использование круглых скобок желательно также в сомнительных случаях для ясности записи и уверенности, что выражение будет вычисляться правильно. Нужно только следить, чтобы количество открывающих скобок совпадало с количеством закрывающих скобок.

Комментарием в REDUCE называется любая последовательность символов: от слова "COMMENT" до ближайшего разделителя (знаков “;” или “\$”) или от символа % до конца строки. Комментарии служат для записи пояснений в программах и системой не обрабатываются.

1.2.2 Числа

В системе REDUCE числа могут быть *целые, рациональные* или *вещественные*.

Целые числа представляют собой последовательности десятичных цифр со знаком или без него, написанных без использования десятичной точки: 88, +5, -7. Они могут входить в любые выражения. Система REDUCE ориентирована на работу с целыми числами и не накладывает ограничений на количество цифр для их представления с произвольной

точностью (см., например, целое число, выражающее результат возведения $2^{**}2000$ и представленное под номером (1.2)).

Рациональные числа представляются в виде отношения двух целых чисел с сокращенными общими множителями. Так обычно представляются дробные числа (в виде отношения двух целых несократимых чисел), *знак* которых хранится в числителе. Поскольку длина целого числа в REDUCE ограничивается лишь размерами используемой оперативной памяти, то это позволяет производить вычисление рациональных значений без округления:

$$\mathbf{aa} := (1/(-2) + 1/7 + 1/11) ** 15; \quad (1.15)$$

$\mathbf{aa} := -1555098314991537910888601 /$
 $649846542388350180836518064717824$

Вещественные числа записываются в двух формах.

- *В основной форме* число записывается в виде целой и дробной частей, разделенных точкой, со знаком +, – или без знака: +151.2, – 0.0001512, 1.512. В большинстве версий REDUCE разрешается использовать не более 8 десятичных цифр.
- *В показательной форме* запись числа состоит из основной формы с указанием десятичного порядка, который обозначается буквой E, а следующая за ней целая константа со знаком или без него определяет величину порядка. Например, +0.1512E3, –0.1512E–3, 151.2E–2.

Предупреждение. Число не может начинаться с точки. Поэтому при записи дробного числа, меньшего единицы, ноль обязательно указывается. Существует мнение, что реализация режима арифметики с плавающей запятой в большинстве версий REDUCE не всегда удачна [7, с. 19].

1.2.3 Идентификаторы

Идентификаторы используются в качестве переменных, меток, а также имен массивов, операций и процедур.

Идентификаторы представляют собой набор букв, цифр и допустимых символов, начинающийся с буквы: N, A1, B2, WORK, KNNEKO. В качестве букв используются буквы латинского алфавита от A до Z, цифры: 0, 1, ..., 9.

Длина идентификатора зависит от конкретной реализации системы (обычно от 1 до 24 символов). Идентификатор нельзя писать в двух строках текста.

Если в идентификатор включается недопустимый символ, то перед таким символом следует употреблять восклицательный знак или закрывающую скобку, что зависит от конкретной реализации системы. Также следует поступать при желании начать идентификатор с цифры.

Однако использование такой возможности нежелательно, так как многие служебные идентификаторы системы REDUCE содержат специальные символы. Их совпадение с идентификатором пользователя может привести к непредсказуемым последствиям. Особенно избегайте использования знака "*", входящего во многие внутренние имена REDUCE.

Запрещено использовать в этом качестве также и *ключевые слова*, означающие имена встроенных в систему команд и процедур. Поэтому они **не должны использоваться в качестве идентификаторов**. Их более восьмидесяти (см. Руководство пользователя по системе REDUCE [25]). Некоторые из них, имеющие 6 и менее символов, приведены ниже:

ALL, ARRAY, BEGIN, BYE, CLEAR, CMD, CONT, CORE, CREATE, DCL, DEFINE, DEPEND, DOWN, EDIT, EMB, END, EXEC, EXPR, FACTOR, FEXPR, FLAGOP, FLOAD, FOR, FORALL, FSLOUT, GO, HELP, IF, IN, INDEX, INPUT, INTEGER, KORDER, LAMBDA, LET, LINEAR, LISP, MACRO, MASS, MATCH, MATRIX, MSHELL, OFF, ON, OPERATOR, ORDER, OUT, PAUSE, PROCEDURE, QUIT, RATIONAL, REAL, REMFAC, REPEAT, RETRY, RETURN, RLISP, SAVEAS, SCALAR, SHARE, SHUT, SLISP, SMACRO, TP, TRST, UP, VECTOR, WEIGHT, WHILE, WRITE, WS.

1.2.4 Переменные

Переменные — это величины, которым присвоены наименования. Для каждой переменной в памяти машины отводится место, в котором хранится ее численное или символьное значение. Для обозначения переменных используются символические имена (идентификаторы).

Каждая переменная обозначается именем, которое является ее первоначальным значением. Необходимо различать имя переменной и ее значение, даже если они совпадают. Имя переменной — обозначение “емко-

сти", которую с помощью операторов присваивания (1.12) наполняют содержимым (численным или символьным значением).

Свободной называется переменная, которой ничего не присвоено и значение которой совпадает с ее именем. *Связанной* — которой ранее было присвоено некоторое значение. Присвоенное переменной значение сохраняется неизменным при дальнейшем выполнении программы или сеанса работы в REDUCE до тех пор, пока не будет изменено оператором присваивания (1.12) или командами CLEAR (1.13), SAVEAS (1.5' ') (или LET). Команда CLEAR делает связанные переменные свободными, какими они считаются в момент начала выполнения программы.

Если в операторе присваивания справа в качестве выражения стоит свободная переменная, то ее *имя присваивается* в качестве вычисленного значения переменной слева:

$$\mathbf{a:=b;}$$
 (1.16)

$a := b$

Если в операторе присваивания справа в качестве выражения стоит связанная переменная, то переменной слева *присваивается вычисленное ранее значение переменной* справа:

$$\mathbf{b:=(x+y*x/2+x*z/3)**3\$}$$
 (1.17)

$$\mathbf{a:=b;}$$

$$a := (x^3*(27*y^3 + 54*y^2*z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216$$

Всегда, когда происходит вычисление выражения, все входящие в него имена исследуются на предмет их совпадения с именами связанных переменных. Если такое совпадение обнаружится, то данное имя заменяется на значение связанной переменной. Имена свободных переменных, которыми являются в нашем примере X, Y и Z, участвуют в дальнейших преобразованиях.

Используются переменные следующих типов: RATIONAL — рациональные, INTEGER, — целые, REAL — вещественные для числовых переменных и SCALAR — для символьных переменных. Рядовой пользователь может не принимать во внимание тип переменных. Если он заранее не объявлен, то по умолчанию переменные имеют тип SCALAR. В качестве значений они принимают представление любого алгебраического выражения. Если оно отсутствует, то они обозначают сами себя.

Некоторые переменные имеют в REDUCE определенные значения:

- E — представляет основание натуральных логарифмов. Если в выражении встречается $\text{LOG}(E)$, то оно автоматически заменяется на единицу. В режиме "ON NUMVAL;" символ E будет заменен его значением с плавающей точкой, округленным с текущей степенью точности арифметики с плавающей запятой.
- I — представляет собой квадратный корень из -1 , I^{*2} заменяется -1 . Аналогично для более высоких степеней I . Используется для представления комплексных чисел.
- PI — число " π ". В режиме "ON NUMVAL;" заменяется значением числа " π " с плавающей точкой и текущей точностью.
- T — имеет значение "истина" и используется в символьном режиме.
- NIL — в алгебраическом режиме работы NIL является синонимом нуля, в символьном режиме имеет значение "ложь".

1.3 ИМЕНА СО СКОБКАМИ

Помимо переменных, описанных выше, в REDUCE используются также так называемые *имена со скобками*, в которых указываются списки параметров, разделенных запятыми: $Q(5)$, $F(X)$, $\text{SIN}(\text{ALFA})$, $\text{COS}(\text{BETA})$, $\text{FY}(A, F(X), \text{SIN}(\text{ALFA}))$. В качестве параметров могут выступать не только простые имена, но и любые выражения, в том числе включающие в себя имена со скобками.

Если список параметров состоит из одного элемента, то скобки, его ограничивающие, можно заменить пробелами. Однако этим нарушается ясность записи. Поэтому мы сами так никогда поступать не станем, и вам не советуем.

Первая операция, которую выполняет REDUCE, встречая в программе имя со скобками, — вычисление значений выражений, записанных в списке параметров. Иначе говоря, *список параметров всегда преобразуется к списку их значений*, и только после этого начинается анализ самого имени со скобками.

В REDUCE имена со скобками применяются для обозначения операторов (см. п. 1.3.1), системных функций (см. п. 1.3.2), элементов массивов (см. п. 1.3.3) и матриц (см. п. 1.3.4), процедур (см. п. 4.6).

1.3.1 Операторы

Функции действующие в системе, называются операторами. Основным свойством оператора является то, что само его имя вместе со списком значений параметров, который у оператора принято называть списком аргументов, может быть его значением. Иными словами, *имя со скобкой, при использовании его в качестве оператора может условно рассматриваться в качестве свободной переменной*. Это свойство свободных имен со скобками в REDUCE при использовании в их качестве системных функций, являющихся также разновидностью операторов, позволит нам в дальнейшем с легкостью проводить исследование вариаций различных геометрических факторов и действующих нагрузок в различных задачах (см. п.).

Оператор следует описать, т.е. указать системе REDUCE, что данное имя со скобками должно ниже обрабатываться как оператор. Для описания операторов используется команда OPERATOR со списком имен, которые далее будут обозначать операторы. Аргументы и скобки у этих имен при описании указывать не надо. Описание каждого оператора проводится один раз в любом месте программы, но до его использования. Например, чтобы считать приведенные в п. 1.3 имена со скобками-операторами, нужно написать:

```
OPERATOR Q, F, FY;
```

Если вышеприведенный оператор отсутствует и в программе встречается никак ранее не описанное имя со скобками, например F(X), то в пакетном режиме при работе с файлами (см. п. 1.4) такое имя автоматически описывается как оператор. При этом выдается сообщение:

```
***F declared operator
```

В интерактивном режиме в подобном случае выдается вопрос:

```
Declare F operator? (Y or N)
```

на который необходимо ответить, набрав либо Y (да), либо N (нет) и нажать клавишу <Enter> – <Ввод>.

При этом *никакого терминатора после букв Y или N не ставится*, поскольку они являются ответами на вопрос, а не предложениями программы.

+В случае утвердительного ответа имя F описывается, как имя оператора, в случае отрицательного — выражение с F(X) игнорируется и выдается сообщение об ошибке в указанном предложении.

1.3.2 Системные функции

В систему REDUCE встроены следующие функции: SIN, COS, TAN, SINH, COSH, TANH, соответствующие тригонометрическим и гиперболическим функциям, и обратные им функции ASIN, ACOS, ATAN, ASINH, ACOSH, ATANH, а также EXP, LOG, SQRT, и некоторые другие (COT, DILOG, ERF, EXPINT).

В качестве своих аргументов эти функции могут использовать только скалярные выражения. Системе известны только элементарные соотношения и свойства этих функций:

$\text{COS}(-X) = \text{COS}(X)$	$\text{SIN}(-X) = -\text{SIN}(X)$
$\text{COS}(N*PI) = (-1)**N$	$\text{SIN}(N*PI) = 0$
$\text{LOG}(E) = 1$	$E**(I*PI/2) = I$
$\text{LOG}(1) = 0$	$E**(I*PI) = -1$
$\text{LOG}(E**X) = X$	$E**(3*I*PI/2) = -I$
$\text{SQRT}(-X) = I*\text{SQRT}(X)$	$\text{SQRT}(A**2) = A$
$\text{SQRT}(200) = 10*\text{SQRT}(2)$	и другие.

Обычно в системе REDUCE происходит замена выражения $\text{SQRT}(A**2)$ на A, которая вполне оправдана, но может повлечь за собой трудности, если выражению A будет случайно присвоено отрицательное значение. Функция извлечения квадратного корня имеет имя SQRT или может быть задана при помощи возведения в степень $** (1/2)$. При выводе печатаются неупрощенные функции извлечения квадратного корня SQRT.

Если установлен режим ON NUMVAL, то функции ACOS, ASIN, COS, EXP, LOG, SIN, SQRT, TAN с числовыми аргументами принимают численные значения в форме с плавающей точкой и текущей степенью точности.

1.3.3 Переменные с индексами. Массивы и оператор ARRAY

Массив — это упорядоченная последовательность величин, называемых элементами массива, обозначаемая одним символическим именем. Каждый элемент массива определяется именем массива и его положением в массиве, т.е. значениями индексов. Индексы заключаются в круглые скобки и разделяются запятыми, причем для двумерного массива первый индекс определяет номер строки, второй — номер столбца. При обозначении массивов пользуются теми же правилами задания типа, что и при обозначении переменных.

Следует отличать размерность массива от его размера:

- *размерность* — число измерений (т.е. число индексов, определяющих положение элемента в массиве);
- *размер* — число его элементов.

Массивы, используемые в программе, должны быть обязательно описаны. Описание массивов дается в начале программы и содержит информацию об именах, размерностях и максимальных размерах используемых массивов. Для описания массивов обычно используется оператор ARRAY, общий вид которого:

```
ARRAY имя_массива  
(верхние_границы_изменения_индексов_через_запятую), ...
```

Индексы в массиве изменяются от 0 до заданной величины. В момент объявления ARRAY все элементы массива обнуляются. Таким образом, переменные, определяющие элементы массива, всегда являются связанными, а значение элемента массива не может совпадать с его именем. В качестве границ массива можно использовать целое положительное выражение.

Не накладывается никаких ограничений на число переменных в одном предложении ARRAY. Он относится к неисполняемым операторам в том смысле, что не порождает команд в программе: по его описанию выделяется место в памяти для размещения всех указанных массивов. Так, после выполнения предложения

```
ARRAY B(6), A(6,6);
```

 (1.18)

в программе могут быть использованы переменные (элементы массива) $B(0), B(1), \dots, B(6)$ и $A(0,0), A(0,1), \dots, A(0,6), A(1,0), A(1,1), \dots, A(1,6), A(2,0), \dots, A(5,6), A(6,0), \dots, A(6,6)$.

Оператор `ARRAY` может появиться в любом месте программы. Если символ объявлен именем массива, он не может быть именем оператора или процедуры. Однако при желании этот символ может быть переобъявлен массивом иной размерности. Отменить описание массива можно командой `CLEAR`, в качестве параметров которой указываются только имена массивов (без скобок с числами):

```
CLEAR A, B; (1.19)
```

Теперь все значения элементов массивов `A` и `B` стираются и перестают занимать память ПК. Сами имена `A` и `B` можно использовать для любых целей, в том числе и для обозначения новых массивов другой длины (см. (1.20)).

1.3.4 Работа с матрицами. Операторы `MATRIX` и `MAT`

Вместо термина “двумерный массив” часто употребляется термин “матрица”. Для работы с матрицами используется оператор `MATRIX`. С его помощью любой идентификатор может быть объявлен матричной переменной, которая обозначает всю матрицу целиком. Размерность матрицы может быть явно задана в описании `MATRIX` после имени матричной переменной. Если размерность неизвестна заранее, ее можно задать неявно, присваивая матричной переменной некоторое матричное выражение. Например, предложение

```
N:=9; MATRIX A(N,N), B(N,1), X(N,1), BB; (1.20)
```

присваивает переменной `N` значение и вводит:

- двумерную матрицу `A` с числом строк `N` и столбцов `N=9`, для которой обеспечивается резервирование памяти для хранения $9 \times 9 = 81$ значения элементов `A`,
- матрицы-столбцы `B` и `X`, содержащих по `N=9` элементов каждый,
- матрицу `BB`, размерность которой определится при присвоении ее элементам каких-либо значений.

В начальном состоянии все элементы некоторой матрицы, размерность которой была описана в операторе `MATRIX`, равны 0. В результате ни один из матричных элементов не может означать сам себя.

Обратим ваше внимание, что индексы у матрицы при использовании операторы MATRIX изменяются от 1 до заданной величины (а не от 0, как у оператора ARRAY), поэтому его удобнее применять при работе с матрицами.

В задачах статики число строк (число уравнений равновесия N), всегда равняется числу столбцов (количеству неизвестных в задаче), поэтому матрица всегда квадратная A (N,N):

$$\begin{array}{l}
 A(1,1), A(1,2), A(1,3), \dots, A(1,N) \\
 A(2,1), A(2,2), A(2,3), \dots, A(2,N) \\
 A(3,1), A(3,2), A(3,3), \dots, A(3,N) \\
 \dots\dots\dots \\
 A(N,1), A(N,2), A(N,3), \dots, A(N,N)
 \end{array} \tag{1.21}$$

Первый индекс обозначает номер строки элемента, а второй — номер столбца. Например, элемент A(2,3) находится во второй строке в третьем столбце.

В операторе MATRIX первое число дает количество строк, а второе количество столбцов.

Замечание. Если матрица введена посредством оператора MATRIX, то сослаться на отдельный элемент этой матрицы или на матрицу можно только после представления размерности матрицы и представления значений ее элементов выполнением матричного присваивания.

Присвоить элементам матрицы некоторые значения можно двумя способами. Рассмотрим их на примере матрицы A(3,3), у которой только элементы главной диагонали отличны от нуля: A(1,1)=1; A(2,2)=2; A(3,3)=3. Для экономии места распечатку представим построчно, а не каждый элемент в отдельной строке, как на экране дисплея.

- По первому из них *присвоение производится только ненулевым элементам* с помощью операторов присваивания. При этом нулевые значения элементов можно не указывать, так как все элементы матрицы, размерность которой была описана, равны 0. Поэтому достаточно только задать ненулевые элементы:

$$\mathbf{matrix\ a(3,3);\ a(1,1):=1;\ a(2,2):=2;\ a(3,3):=3;\ a;} \tag{1.22}$$

$$\begin{array}{lll}
 \mathbf{mat(1,1) := 1} & \mathbf{mat(1,2) := 0} & \mathbf{mat(1,3) := 0} \\
 \mathbf{mat(1,1) := 0} & \mathbf{mat(1,2) := 2} & \mathbf{mat(1,3) := 0} \\
 \mathbf{mat(1,1) := 0} & \mathbf{mat(1,2) := 0} & \mathbf{mat(1,3) := 3}
 \end{array} \tag{1.23}$$

- Другой способ присвоения — это *использование оператора MAT*, в котором *элементы матрицы задаются в полном виде по строкам*, каждая из которых заключена во внутренние скобки.

$$\mathbf{matrix\ a(3,3);\ a:=mat((1,0,0), (0,2,0), (0,0,3));\ a; \quad (1.24)}$$

Результаты выполнения предложения (1.24) полностью совпадают с (1.23), потому повторно не приводятся.

Отметим, что *аргументы оператора MAT обязательно должны заключаться в скобки*, даже если представляется матрица, содержащая 1 столбец или 1 элемент.

Поэтому матрица-столбец задается в виде MAT ((X), (Y)). Внутренние скобки нужны здесь, чтобы отличать этот случай от случая матрицы-строки, которая задается в виде MAT (X,Y).

Матричные выражения. Весьма впечатляющей особенностью системы REDUCE является та легкость, с которой могут быть выполнены матричные вычисления.

Они следуют обычным правилам матричной алгебры. Это позволяет легко решать системы линейных алгебраических уравнений в численном и символьном виде (см. п. 2), а также проводить любые преобразования с использованием матричных переменных [24, с. 140-144].

При этом следует помнить, что в суммах и произведениях матричных выражений размерности должны быть приведены в соответствие.

1.4 КОМАНДЫ РАБОТЫ С ФАЙЛАМИ. СОЗДАНИЕ, РЕДАКТИРОВАНИЕ И ИСПОЛЬЗОВАНИЕ ФАЙЛОВ.

Система REDUCE создавалась как интерактивная система. Однако она может работать и в пакетном режиме с предварительно созданными файлами редьюсовских программ.

Это удобнее по многим причинам:

- программы можно предварительно внимательно проверить;
- результат их выполнения также можно записать в файл на диск;
- после первоначального выполнения можно неоднократно доводить программу до правильного решения, внося необходимые изменения;

- создание и редактирование файлов можно выполнять с помощью практически любого текстового редактора, которые гораздо удобнее встроенного редьюсовского;
- после доведения решения до правильного можно исследовать влияние вариаций различных факторов на базе легко получаемых копий этого файла;

Для работы с файлами в системе REDUCE имеется несколько команд: OUT, SHUT и IN.¶

1.4.1 Команды OUT и SHUT

В качестве параметра команды OUT может быть идентификатор единственного файла, заключенный в кавычки в большинстве версий REDUCE:

```
OUT "C:\REDUCE\PROG\PR3.LIS"; (1.25)
```

Эта команда дает указание системе, что с места ее расположения в программе все результаты следует направлять в указанный файл C:\REDUCE\PROG\PR3.LIS до тех пор, пока другая команда OUT не изменит файл вывода или команда SHUT не закроет его.

Напомним, что идентификатор файла полностью описывает расположение файла на диске и имеет следующий формат:

```
[диск:] [путь\]имя_файла[.расширение] (1.26)
```

Здесь элементы, заключенные в квадратные скобки, являются необязательными. В случае, если в описании не указывается дисковод или путь, то подразумевается текущий дисковод или текущий каталог. Идентификатором находящегося в нем файла ИДФ будет только имя файла с его расширением (если оно имеется).

Последняя возможность очень удобна. Поэтому для файла, находящегося в текущем каталоге, предложение (1.25) примет вид:

```
OUT "PR3.LIS"; (1.27)
```

Одновременно для вывода могут быть открыты несколько файлов. Однако вывод будет производиться только в файл, открытый последним:

```
OUT "FR1.L"; {команды REDUCE};
```

```
OUT "FR2.L"; {команды REDUCE};
```

OUT T; {команды REDUCE};

OUT "FR1.L";

- после команды OUT "FR1.L"; вывод будет происходить в файл "FR1.L";
- после OUT "FR2.L"; — в файл "FR2.L";
- по OUT T; —направляться на терминал (дисплей);
- после второй команды OUT "FR1.L" вывод будет происходить в конец файла "FR1.L".

Результаты, посылаемые в файл, будут иметь такой же вид, какой они имели бы на экране дисплея. По умолчанию в стандартных математических обозначениях (установлен флаг ON NAT;). Например, X^{**2} будет записано как X^2 .

Если целью вывода является *сохранение результатов для их дальнейшего использования* в системе REDUCE, то такая форма является неприемлемой.

Поэтому мы должны перед созданием файла установить переключатель в состояние OFF NAT.

После этого в конце любого выражения в качестве ограничителя появится знак "\$", а сама информация будет записана в виде, позволяющем использовать ее в системе REDUCE для последующего ввода.

Далее в конце файла следует ввести слово END командой WRITE “;END”\$, что является стандартным способом заканчивать файлы, предназначенные для считывания.

При любой форме вывода результатов все открытые файлы следует закрыть командой SHUT, после которой в кавычках пишется список идентификаторов файлов, которые были открыты командой OUT. По команде SHUT эти файлы будут закрыты:

SHUT "PR3.LIS"; (1.28)

При выходе из системы REDUCE с файлом, незакрытым командой SHUT, может произойти потеря информации.

Если же файл закрыт командой SHUT и выдается команда OUT для этого файла, то он стирается, а затем записывается новый выходной файл. Попытки закрыть файлы, которые не были открыты командой OUT, приводят к возникновению ошибки.

При работе в интерактивном режиме при обнаружении неверно записанного выражения (синтаксической ошибки) выдается сообщение об ошибке, поэтому пользователь имеет возможность ввести вместо неправильно записанного предложения правильное.

В пакетном режиме при обнаружении ошибки дальнейшее выполнение программы сводится лишь к проверке на правильность записи (синтаксическому анализу), но никаких вычислений после этого не производится.

Пример. Последовательность команд

```
off nat; out "vivf.l"; k:=(c+d)**4; write ";end";  
shut "vivf.l"; on nat;
```

позволит сформулировать выводной файл VIVF.L, который будет содержать запись информации в виде, позволяющем использовать ее для последующего ввода:

```
k := c**4 + 4*c**3*d + 6*c**2*d**2 + 4*c*d**3 + d**4$  
;end$
```

Отметим, что сохранение результатов для их дальнейшего использования может быть также выполнено в синтаксисе Фортрана с помощью флага FORT (см. дополнение 3.3).

1.4.2 Команда IN

В качестве аргументов этой команды могут быть допустимые имена файлов.

```
IN "PR3", "C:\REDUCE\PROG\PR4 "; (1.29)
```

По этой команде в систему будет загружен файл PR3, а затем C:\REDUCE\PROG\PR4.

Если команда IN ограничена символом ";", то содержимое файла отображается на экране или в выходном файле. Чтобы предотвратить указанное отображение, необходимо использовать либо ограничитель "\$" в IN, либо во входном файле команду OFF ECHO;

Файлы, считываемые по команде IN, обычно завершаются конструкцией

```
;END; (1.30)
```

Знак “точка с запятой” перед словом END используется для большой надежности, хотя его можно и не указывать. Он запускает специальный учет контроля за файлами, улучшающий эффективность работы системы, а также защищает от некоторых ошибок при использовании сложных составных операторов. Если же слово END пропущено, то выдается сообщение об ошибке.

Замечание. Работать в текущем каталоге очень удобно: все команды открытия (1.27), закрытия (1.28) или работы с файлами (1.29) предстают в своей самой простой форме. Однако текущим каталогом при работе с системой REDUCE будет сам подкаталог REDUCE, ибо из него в начале всегда запускается сама система. Работать в подкаталоге в окружении системных файлов очень неудобно, ибо всегда есть опасность удалить или запортить один из них с непредсказуемыми последствиями. Поэтому в дальнейшем *будем организовывать для работы подкаталог в корневом каталоге текущего диска*, например \P-R, в котором будут располагаться сами программы и результаты их работы. В этом случае предложения (1.27), (1.28) и (1.29) соответственно примут вид (1.31) – (1.33):

OUT "\P-R\PR3.LIS"; (1.31)

SHUT "\P-R\PR3.LIS"; (1.32)

IN "\P-R\PR3", "\P-R\PR4 "; (1.33)

Пример. Чтобы команды (1.3') – (1.10') из п. 1.1 выполнить из внешнего файла, например SR1, расположенного в подкаталоге \P-R корневого каталога текущего диска, а результат их выполнения записать на диск в файл SR1.LIS в том же подкаталоге, нужно:

- создать в корневом каталоге текущего диска (где находится и подкаталог \REDUCE с расположенными в нем системными файлами reduce.exe или reduce.bat) подкаталог с нужным именем, например, P-R. Для этого в корневом каталоге текущего диска следует нажать функциональную клавишу <F7> и в выводимой на экран строке следует указать имя нового подкаталога (например, P-R). После нажатия клавиши <Enter> в оглавлении текущего каталога появится новый пустой подкаталог с указанным именем P-R. После этого нужно совместить курсор с его именем и нажать клавишу <Enter> (<ВВОД>). Norton Commander (Volkov Commander или другая аналогичная оболочка) “войдет” в этот подкаталог;

- в подкаталоге \P-R корневого каталога текущего диска создать в любом текстовом редакторе новый файл с нужным именем, например SR1. Для этого необходимо при любом положении курсора нажать клавиши <Shift>+<F4> (<Верх>+<Ф4>). Затем в появившемся на экране приглашении набрать имя файла (например, SR1) и нажать клавишу <Enter>. Если файл с таким именем существует, то он вызывается на редактирование, если нет — то откроется пустой файл;
- начать набор исходных данных:
 - в первой команде OUT в качестве параметра указать идентификатор открываемого для вывода файла результатов \P-R\SR1.LIS (например, OUT "\P-R\SR1.LIS");
 - далее в удобной форме (в одной или нескольких строках) набрать последовательность команд (1.3') — (1.10') из п. 1.1 (выделенных жирным шрифтом);
 - в предпоследней команде SHUT в качестве параметра указать идентификатор закрываемого файла результатов \P-R\SR1.LIS (например, SHUT "\P-R\SR1.LIS");
 - в конце следует поместить команду END;
 - после окончания ввода команд, проверки набранной информации и исправления ошибок, ее необходимо сохранить. Для этого нужно нажать клавишу <F2> (<Ф2>) для записи на диск с именем редактируемого файла \P-R\SR1. Для выхода из режима редактирования нужно нажать клавишу <F10> (<Ф10>). На экране опять появляются панели Norton Commander.

В результате файл SR1 примет, например, следующий вид:

```

OUT "\P-R\SR1.LIS"; TT:=(X+Y*X/2+X*Z/3)**3; OFF ALLFAC;
TT; TTD:=DF(TT,X); ON ALLFAC; TTI:=INT(TTD,X);           (1.34)
ON FORT; TTI; OFF FORT; SHUT "\P-R\SR1.LIS"; ;END;

```

После этого нужно выйти из подкаталога \P-R и перейти в подкаталог \REDUCE с расположенными в нем системными файлами. Затем следует загрузить систему, для чего нужно запустить на выполнение файл reduce.exe или reduce.bat, совместив с их именами подсветку курсора и нажав клавишу <Enter> (<Ввод>). После появления приглашения в командной строке следует ввести:

1: in "\P-R\SR1"\$ (1.35)

По этой команде в систему будет загружен файл \P-R\SR1, выполнены все находящиеся в нем команды (1.34), а результаты их выполнения будут записаны на диске в файле SR1.LIS в том же подкаталоге \P-R текущего диска.

Если в качестве ограничителя оператора IN (1.35) используется:

- **точка с запятой**, то операторы, содержащиеся в файле (для нашего примера \P-R\SR1), вместе с результатами их работы записываются в текущий файл вывода (\P-R\SR1.LIS);
- **знак \$**, то сами операторы файла ввода (\P-R\SR1) не распечатываются, а в файле вывода (\P-R\SR1.LIS) записываются только результаты их работы.

В обоих случаях для просмотра результатов работы программы из файла \P-R\SR1 (1.34):

- следует выйти из сеанса работы с REDUCE по команде **bye**;
- выйти из подкаталога \REDUCE и перейти в подкаталог \P-R корневого каталога текущего диска;
- совместить подсветку курсора с именем файла результатов SR1.LIS и нажать клавишу <F3> (<Ф3>).

Результаты работы программы будут иметь для нашего примера следующий вид, соответственно совпадающий с выражениями из п. 1.1 (1.4'), (1.6') — (1.9').

$$tt := (x^3*(27*y^3 + 54*y^2 *z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216$$

$$(27*x^3 *y^3 + 54*x^3 *y^2 *z + 162*x^3 *y^2 + 36*x^3 *y*z^2 + 216*x^3 *y*z + 324*x^3 *y + 8*x^3 *z^3 + 72*x^3 *z^2 + 216*x^3 *z + 216*x^3)/216$$

$$ttd := (27*x^2 *y^3 + 54*x^2 *y^2 *z + 162*x^2 *y^2 + 36*x^2 *y*z^2 + 216*x^2 *y*z + 324*x^2 *y + 8*x^2 *z^3 + 72*x^2 *z^2 + 216*x^2 *z + 216*x^2)/72$$

$$tti := (x^3*(27*y^3 + 54*y^2 *z + 162*y^2 + 36*y*z^2 + 216*y*z + 324*y + 8*z^3 + 72*z^2 + 216*z + 216))/216$$

$$ans = (x**3*(27.*y**3+54.*y**2*z+162.*y**2+36.*y*z**2+216.*y*z+324.*y+8.*z**3+72.*z**2+216.*z+216.))/216.$$

Замечание. Иногда при дальнейшем использовании файла в интерактивном режиме нам требуется ввести дополнительные данные (например,

присвоить значения некоторой переменной или ввести некоторую функцию) или провести некоторые вычисления до завершения программы. Тогда в соответствующем месте файла (программы) нужно вставить команду PAUSE. При последующем использовании этого файла система останавливает выполнение программы и печатает на терминале сообщение CONT? . Если мы хотим продолжить выполнение программы, то следует ввести Y (соответствует английскому слову yes). Для временной остановки выполнения программы следует напечатать N (соответствует английскому слову no). Во время этой остановки пользователь может вводить с терминала требуемые команды и выражения, а также выполнять требуемые вычисления, после чего нужно ввести команду CONT; и выполнение программы продолжится.

Если вы не испытываете затруднений при общении с компьютером, то можно сразу перейти к изучению п.2. В противном случае быстрый старт несколько затянется и вам следует подробнее изучить нужные места части 2-й.

до медленнее, чем непосредственное вычисление выражения $(1/A)*B$, так как в этом случае система в целом обрабатывает гораздо меньшее количество элементов.

2.1 ЭТАПЫ РЕШЕНИЯ СЛАУ И СТРУКТУРА ПРОГРАММ

Для решения на ПК любой СЛАУ, возникающей в задачах статики, ее сначала нужно формализовать, приведя к виду (2.2).

Это достигается постановкой в соответствие неизвестным в задаче (усилиям в стержнях, опорным реакциям, неизвестным силам и т.п.) определенного элемента одномерного массива $X(I)$ (см. соответствие идентификаторов в п.п. 2.2, 2.3, 4).

Формализация помогает как правильному представлению полного вида матрицы A (2.2) для ее задания с помощью оператора MAT, так и заданию только ненулевых ее элементов с помощью операторов присваивания.

Вторым этапом следует численное или символьное определение всех коэффициентов при неизвестных в уравнениях равновесия и их свободных членов (не содержащих неизвестных). Последние переносятся в правую часть этих уравнений, образуя матрицу-столбец свободных членов $B(I)$.

После выполнения 2-го этапа система уравнений любой задачи статики по определению реакций опор приобретает явный вид СЛАУ (2.2), в которой выписаны только отличные от нуля элементы.

Для дальнейшего решения с помощью ПК в системе имеются две возможности составления программ:

- ввод всех элементов матрицы $A(I,J)$ и матрицы-столбца $B(I,1)$ для использования оператора MAT, что требует представления полного вида матриц с помощью добавления соответствующего количества нулевых элементов;
- ввод только ненулевых элементов с указанием их расположения, что требует использования операторов присваивания.

Соответственно этому *структура простейшей программы* для решения задач статики в системе REDUCE состоит из следующих основных блоков:

- открытие командой OUT выводного файла с идентификатором ИДФ (1.26), куда будут записываться результаты работы программы (OUT "ИДФ";)
- задание значений с помощью операторов присваивания используемым вспомогательным переменным и тригонометрическим функциям (N:=6;)
- включение или выключение соответствующих переключателей, устанавливающих нужные режимы печати и представления результатов:
 - включение режима FLOAT при работе с вещественной арифметикой (ON FLOAT;) или выключение его при использовании рациональных чисел (OFF FLOAT;),
 - запрещение печати нулевых значений при контрольной распечатке матриц A и B, что достигается включением флага NERO (ON NERO;)
- описание матричных переменных оператором MATRIX с явным заданием ее размерностей

MATRIX A(N,N), B(N,1), X(N,1);

- ввод исходных данных одним из двух способов:
 - с помощью оператора MAT, в котором элементы матрицы задаются в полном виде по строкам, каждая из которых заключена во внутренние скобки

A:=MAT((,...,), (по строкам во внутренних скобках все элементы через запятую), (,...,));

B:=MAT((), ... (по элементу во внутренних скобках), ());

- использование операторов присваивания для ввода только нулевых значений соответствующих элементов матрицы A(I,J) и матрицы-столбца B(I,1) с указанием их расположения (см. дополнение 2.1);
- контрольная печать введенных исходных данных автоматически получается применением терминатора ";" при любом способе ввода и может быть в двух формах в зависимости от используемой версии REDUCE (по этой причине она в рассматриваемых примерах не приводится, но является очень важной для проверки правильности введенных в ЭВМ исходных данных):

- представление матриц с указанием номера соответствующего элемента и его значения по одному элементу в каждой строке (версии до REDUCE 3.3 включительно),
- распечатка значений элементов матриц в их естественной форме по строкам без указания номера соответствующего элемента (версия REDUCE 3.5), при этом не работает флаг NERO;
- разрешение печати нулевых значений матрицы-столбца результатов решения X , что достигается выключением флага NERO (OFF NERO;)
- непосредственное решение СЛАУ задачи статики и печать выходной информации (значений элементов матрицы-столбца $X(I,1)$), которая автоматически получается применением терминатора ";":

$X := A^{**}(-1) * B;$

- закрытие командой OUT выводного файла с ИДФ (1.26), куда записывались результаты работы программы (SHUT "идф";)
- в конце следует ввести оператор END, что является стандартным способом заканчивать файлы, предназначенные для считывания (END; или для большей надежности ;END;).

Для удобства изложения можно разбить все рассматриваемые типовые примеры по формальному признаку количества уравнений равновесия на три группы: с малым (до 6-ти — см. п. 2.2.1), средним (до 12-ти — см. п. 2.3) и большим (свыше 12-ти — см. п. 2.2.2) числом уравнений. Отметим, что достоинства использования операторов присваивания для ввода только ненулевых элементов резко возрастают с увеличением числа уравнений равновесия.

Рассмотрим решение задач статики с использованием CAB REDUCE на типовых примерах, приведенных в сборнике [22], которым присвоен 31-й вариант.

2.2 РЕШЕНИЕ ЗАДАЧ СТАТИКИ НА ПК В САВ REDUCE С “МАЛЫМ” КОЛИЧЕСТВОМ УРАВНЕНИЙ

2.2.1 Формализация уравнений равновесия

В качестве примера задачи статики с “малым” количеством уравнений равновесия воспользуемся аналитическим решением типового задания С-8 по определению усилий в шести стержнях пространственной шарнирно-стержневой конструкции для ССС [22, с. 52-54]. Его расчетная схема приведена на рис. 2.1.

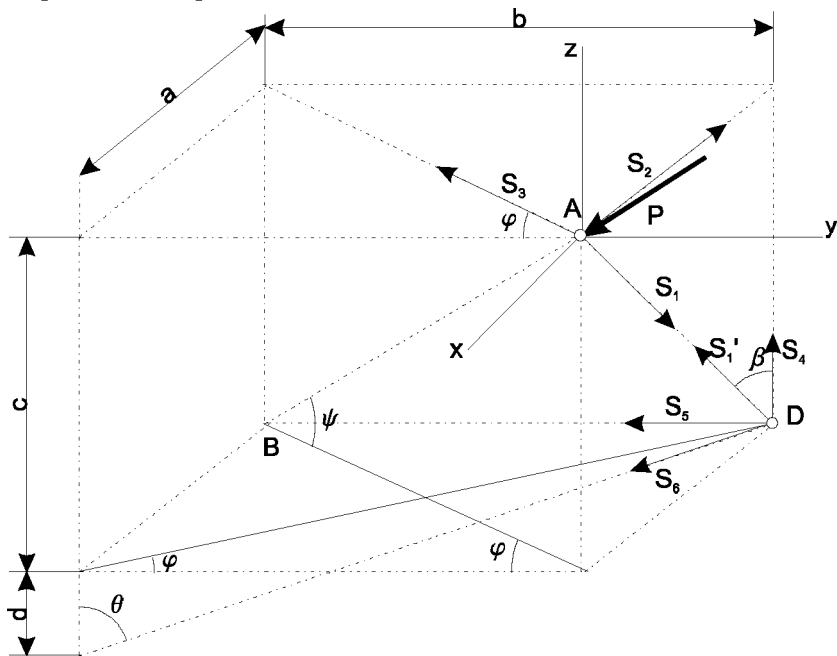


Рис. 2.1. Расчетная схема пространственной шарнирно-стержневой конструкции

Составим по три уравнения равновесия для сил, сходящихся соответственно в узлах А и D (они приведены под номерами (1)–(6) в пособии [22, с. 52-53]):

$$\begin{aligned}
\Sigma X_i = 0; & \quad -P \cdot \cos \psi \cdot \sin \varphi - S_1 \cdot \sin \beta - S_2 - S_3 \cdot \sin \varphi = 0, \\
\Sigma Y_i = 0; & \quad -P \cdot \cos \psi \cdot \cos \varphi - S_3 \cdot \cos \varphi = 0, \\
\Sigma Z_i = 0; & \quad -P \cdot \sin \psi - S_1 \cdot \cos \beta = 0, \\
\Sigma X_i = 0; & \quad S_1' \cdot \sin \beta + S_6 \cdot \sin \theta \cdot \sin \varphi = 0, \\
\Sigma Y_i = 0; & \quad -S_5 - S_6 \cdot \sin \theta \cdot \cos \varphi = 0, \\
\Sigma Z_i = 0; & \quad S_1' \cdot \cos \beta + S_4 - S_6 \cdot \cos \theta = 0.
\end{aligned} \tag{2.3}$$

Система уравнений равновесия (2.3) является системой линейных алгебраических уравнений, в которой выписаны только отличные от нуля значения элементов матрицы $A(I, J)$ в неформализованном виде.

Рассмотрим подготовку исходных данных для решения этого примера на ПК в САВ REDUCE.

Отметим, что внутренние силы в первом стержне S_1 и S_1' при мысленном его разбиении для рассмотрения равновесия узлов А и D становятся внешними для каждого из них, выражая действие отброшенной части. Они являются силами действия и противодействия, их модули равны друг другу и они противоположны по направлению. Этот факт выражают векторные равенства:

$$\vec{s}_1 = -\vec{s}_1'. \tag{2.4}$$

Обычно векторные уравнения (2.4), выражающие равенство сил действия и противодействия, учитывают на рисунке, направляя соответствующие вектора для штрихованных и не штрихованных величин в противоположные стороны, как это показано на рис. 53 [22, с.53].

Так как разность направлений штрихованных и не штрихованных величин уже учтена на рисунках (т.е. учтен знак минус в уравнении (2.4)), то этому векторному выражению (2.4) соответствует алгебраическое уравнение (2.5), учитывающее равенство значений этих величин вплоть до знака, так что им можно присвоить один идентификатор:

$$s_1 = s_1' = X_1 \tag{2.5}$$

Эти равенства (2.5) используют при решении задачи. Такой подход мы рекомендуем применять всегда не только при определении усилий в стержнях ферм в п. 5.2, но и при рассмотрении равновесия систем тел при

направлении внутренних сил в односторонних связях в п. 6 и промежуточных шарнирах (когда в узле сходятся не более двух стержней).

Соответствие идентификаторов для рассматриваемого типового примера с учетом вышеизложенного примет вид:

$$\begin{array}{cccccc}
 X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \\
 S_1=S_1' & S_2 & S_3 & S_4 & S_5 & S_6
 \end{array} \quad (2.6)$$

Для удобства составления программ заменим также используемые греческие буквы в обозначениях углов их идентификаторами, записанными в их латинской транскрипции:

$$\begin{array}{cccc}
 \text{BETA} & \text{FI} & \text{TETA} & \text{PSI} \\
 \beta & \varphi & \theta & \psi
 \end{array} \quad (2.7)$$

Перенесем свободные члены, не содержащие неизвестных, в правые части уравнений (2.3), которые с учетом соответствия идентификаторов (2.6)–(2.7) теперь примут следующий формализованный вид (2.8):

$$\begin{aligned}
 -X_1 \sin(\text{BETA}) - X_2 - X_3 \sin(\text{FI}) &= P \cos(\text{PSI}) \sin(\text{FI}), \\
 -X_3 \cos(\text{FI}) &= P \cos(\text{PSI}) \cos(\text{FI}), \\
 -X_1 \cos(\text{BETA}) &= P \sin(\text{PSI}), \\
 X_1 \sin(\text{BETA}) + X_6 \sin(\text{TETA}) \sin(\text{FI}) &= 0, \\
 -X_5 - X_6 \sin(\text{TETA}) \cos(\text{FI}) &= 0, \\
 X_1 \cos(\text{BETA}) + X_4 - X_6 \cos(\text{TETA}) &= 0.
 \end{aligned} \quad (2.8)$$

Теперь система уравнений (2.8) приобрела явный вид СЛАУ (2.2), в которой выписаны только отличные от нуля элементы. Для дальнейшего её решения или исследования с помощью ПК в системе REDUCE имеются различные возможности.

2.2.2 Решение в виде вещественных и рациональных чисел

Перед составлением программы для численного решения системы уравнений (2.8) в виде вещественных чисел по заданным размерам

($a = 4$ м, $b = 5$, $c = 4$, $d = 1$ м) вычисляем тригонометрические функции используемых углов [см. 22, с. 52-54]:

$$\sin\varphi = \sin(\text{FI}) = \frac{a}{\sqrt{a^2 + b^2}} = \frac{4}{\sqrt{4^2 + 5^2}} = \frac{4}{\sqrt{41}} = 0.6247;$$

$$\cos\varphi = \cos(\text{FI}) = \frac{b}{\sqrt{a^2 + b^2}} = \frac{5}{\sqrt{4^2 + 5^2}} = \frac{5}{\sqrt{41}} = 0.7809;$$

$$\sin\psi = \sin(\text{PSI}) = \frac{c}{\sqrt{a^2 + b^2 + c^2}} = \frac{4}{\sqrt{4^2 + 5^2 + 4^2}} = \frac{4}{\sqrt{57}} = 0.5298;$$

$$\cos\psi = \cos(\text{PSI}) = \frac{\sqrt{a^2 + b^2}}{\sqrt{a^2 + b^2 + c^2}} = \frac{\sqrt{4^2 + 5^2}}{\sqrt{4^2 + 5^2 + 4^2}} = \frac{\sqrt{41}}{\sqrt{57}} = 0.8481;$$

$$\sin\beta = \sin(\text{BETA}) = \frac{a}{\sqrt{a^2 + c^2}} = \frac{4}{\sqrt{4^2 + 4^2}} = \frac{1}{\sqrt{2}} = 0.7071; \quad (2.9)$$

$$\cos\beta = \cos(\text{BETA}) = \frac{b}{\sqrt{a^2 + c^2}} = \frac{4}{\sqrt{4^2 + 4^2}} = \frac{1}{\sqrt{2}} = 0.7071;$$

$$\sin\theta = \sin(\text{TETA}) = \frac{\sqrt{a^2 + b^2}}{\sqrt{a^2 + b^2 + d^2}} = \frac{\sqrt{41}}{\sqrt{4^2 + 5^2 + 1^2}} = \frac{\sqrt{41}}{\sqrt{42}} = 0.988;$$

$$\cos\theta = \cos(\text{TETA}) = \frac{d}{\sqrt{a^2 + b^2 + d^2}} = \frac{1}{\sqrt{4^2 + 5^2 + 1^2}} = \frac{1}{\sqrt{42}} = 0.1543.$$

После численного определения всех коэффициентов при неизвестных и вычисления при значении $P = 4$ свободных членов (не содержащих неизвестных), уравнения (2.8) примут формализованный вид (2.10):

1. $-0.7071 \cdot X_1 - X_2 - 0.6247 \cdot X_3 = 2.1192$,
2. $-0.7809 \cdot X_3 = 2.6491$,
3. $-0.7071 \cdot X_1 = 2.1192$,
4. $0.7071 \cdot X_1 + 0.6172 \cdot X_6 = 0$,
5. $-X_5 - 0.7715 \cdot X_6 = 0$,
6. $0.7071 \cdot X_1 + X_4 - 0.1543 \cdot X_6 = 0$.

(2.10)

Для удобства пользования уравнения пронумерованы сверху вниз, начиная с 1, где номером без скобок обозначается порядковый номер строки в системе уравнений (2.10).

Теперь система уравнений (2.10) приобрела явный вид СЛАУ (2.2), в которой выписаны в вещественной форме только отличные от нуля элементы. Поэтому квадратную матрицу А для ее представления в операторе MAT следует дополнить нулями до ее полного вида $N \times N$:

$$\begin{aligned}
 &0.7071, -1, -0.6247, 0, 0, 0 \\
 &0, 0, -0.7809, 0, 0, 0 \\
 &-0.7071, 0, 0, 0, 0, 0 \\
 &0.7071, 0, 0, 0, 0, 0.6172 \\
 &0, 0, 0, 0, -1, -0.7715 \\
 &0.7071, 0, 0, 1, 0, -0.1543
 \end{aligned} \tag{2.11}$$

Структура простейшей программы в системе REDUCE из п. 2.1 для решения системы уравнений (2.10) может быть реализована, например, следующим образом (по вопросам записи программы 2.1, например, в файле PR2-1 в подкаталоге \C8 корневого каталога текущего диска и запуске ее на выполнение см. п. 7 и пример в п. 1.4.2):

COMMENT **Программа 2.1:** решение типового примера С-8 на REDUCE с заданием с помощью оператора MAT всех элементов матрицы А и матрицы-столбца В с представлением данных ввода и результатов в численном виде;

```

OUT "\C8\PR2-1.LIS";          10
ON FLOAT, NERO;              15
N:=6;                        20
MATRIX A(N,N), B(N,1), X(N,1); 30
A:=MAT((-0.7071,-1,-0.6247,0,0,0), 35
(0,0,-0.7809,0,0,0),
(-0.7071,0,0,0,0,0),
(0.7071,0,0,0,0,0.6172),
(0,0,0,0,-1,-0.7715),
(0.7071,0,0,1,0,-0.1543));
B:=MAT((2.1192),(2.6491),(2.1192),(0),(0),(0)); 55
OFF NERO;                    80
X:=A**(-1)*B;                85
OFF FLOAT;                   90
SHUT "\C8\PR2-1.LIS";       95
;END;                         99

```

Номера справа операторов и команд проставлены только для удобства дальнейших пояснений. В программе на экране дисплея *операторы не нумеруются*. Этот прием будет использоваться и в дальнейшем при описании программ.

Также соображения удобства пояснений определили расположение почти каждого оператора или команды программы 2.1 в отдельной строке.

Напомним, что их может быть как несколько в одной строке записи (см., например, оператор 15, что будет использоваться нами в дальнейшем при описании однотипных операторов), так и каждый из них может занимать несколько строк, при этом лишние пробелы игнорируются и никаких знаков переноса не используется (см., например, оператор 35).

Комментарий, помещенный между зарезервированным словом COMMENT и разделителем в виде знака “точка с запятой”, описывает назначение программы 2.1 и при обработке игнорируется. В комментариях также обычно указываются номера группы, задания и варианта, а также фамилия и инициалы автора данной программы.

Команда 10 открывает выводной файл PR2-1.LIS, находящийся в подкаталоге \C8 корневого каталога текущего диска, куда будут записываться результаты работы программы 2.1. Отметим, что имя файла и подкаталога для ваших программ может быть любым разрешенным идентификатором и должно отражать индивидуальность вашу (для файла) или группы (для подкаталога).

Команда 15 включает режим FLOAT для работы с вещественной арифметикой и запрещает печать нулевых значений при контрольной распечатке матриц A и B, что достигается включением флага NERO.

Оператор присваивания 20 задает значение используемой вспомогательной переменной N, равное количеству шести уравнений равновесия в данной задаче.

Оператор 30 описывает матричные переменные с явным заданием их размерностей:

- матрицу A с числом строк N и столбцов N=6, для которой обеспечивается резервирование памяти для хранения $6 \times 6 = 36$ значений элементов A,
- матрицы-столбцы B и X, содержащих по N=6 элементов каждый.

Ввод исходных данных матрицы А и матрицы-столбца В осуществляется с помощью операторов 35 и 55 соответственно. Элементы матриц в форме вещественных чисел задаются в полном виде по строкам, каждая из которых заключена во внутренние скобки (даже если она состоит из одного элемента), как это требуется для оператора MAT. Контрольная печать введенных исходных данных будет получена автоматически применением терминатора ";".

Команда 80 разрешает печать нулевых значений перед решением СЛАУ, что достигается выключением флага NERO. Это сделано для возможности распечатки всех значений матрицы-столбца X.

Оператор 85 выполняет непосредственное решение СЛАУ задачи статики и печать выходной информации (значений элементов матрицы-столбца X(I,1)).

Команда 90 выключает режим работы с вещественной арифметикой, что достигается выключением флага FLOAT.

Команда 95 закрывает выводной файл PR2-1.LIS, находящийся в подкаталоге \C8 корневого каталога текущего диска, куда записывались результаты работы программы 2.1.

Оператор 99 запускает специальный учет контроля за файлами, улучшающий эффективность работы системы, что является стандартным способом заканчивать файлы, предназначенные для считывания. Знак "точка с запятой" перед словом END используется для большей надежности, хотя его можно и не указывать. В дальнейшем мы будем применять его только при использовании сложных составных операторов.

Результаты решения после окончания работы программы 2.1 (вместе со всей предусмотренной в ней печатью) окажутся в файле PR2-1.LIS, находящемся в подкаталоге \C8 корневого каталога текущего диска, и будут иметь для рассматриваемого примера следующие значения:

$$\begin{array}{ll} X(1,1) := -2.99701 & X(2,1) := 2.119203 \\ X(3,1) := -3.39236 & X(4,1) := 2.648988 \\ X(5,1) := -2.64898 & X(6,1) := 3.43355 \end{array} \quad (2.13)$$

Расположение элементов матриц на экране дисплея, как отмечалось в п. 2.1, будет зависеть от используемой версии REDUCE.

Дополнение 2.1. *Положение элемента $A(I,J)$ в матрице A характеризуется двойным индексом. Первый индекс I означает номер строки, в которой стоит элемент $A(I,J)$ – это порядковый номер уравнения равновесия. Второй индекс J означает номер столбца – номер идентификатора X_j , при котором стоит элемент $A(I,J)$.* Отметим, что нумерация строк производится сверху вниз, а столбцов — слева направо.

Для матрицы A ненулевыми элементами $A(I,J)$ системы уравнений (2.10) будут:

- для 1-й строки (уравнение 1.) первый индекс $I=1$, второй индекс J совпадает с номером идентификатора, при котором стоит значение элемента $A(1,J)$: $A(1,1)=-0.7071$ ($J=1$, так как значение этого элемента, равное -0.7071 , стоит при идентификаторе X_1), $A(1,2)=-1$ ($J=2$, так как значение элемента $A(1,2)$ стоит при идентификаторе X_2), $A(1,3)=-0.6247$ ($J=3$, так как стоит при X_3). Напомним, что все остальные элементы первой строки будут равны нулю: $A(1,4)=A(1,5)=A(1,6)=0$.
- для 2-й строки (уравнение 2.) первый индекс $I=2$, второй индекс совпадает с номером идентификатора, при котором стоит значение элемента $A(2,J)$: $A(2,3)=-0.7809$ (значение стоит при X_2 , значит второй индекс $J=2$).
- для 3-й строки (уравнение 3.): $A(3,1)=-0.7071$.
- для оставшихся строк 4-6 аналогично: $A(4,1)=0.7071$, $A(4,6)=0.6172$, $A(5,5)=-1$, $A(5,6)=-0.7715$, $A(6,1)=0.7071$, $A(6,4)=1$, $A(6,6)=-0.1543$.

Напомним, что в каждой строке по $N=6$ элементов, но значения всех остальных невыписанных элементов равны нулю.

Положение элемента вектора-столбца $V(I)$ характеризуется только номером I -й строки, в которой стоит элемент $V(I)$. При представлении его в REDUCE в виде матрицы-столбца это примет вид $V(I,1)$, где 1 – номер единственного первого столбца.

Для матрицы-столбца V ненулевыми элементами единственного первого столбца $V(I,1)$ системы уравнений (2.10) будут:

- для 1-й строки $I=1$ и $V(1,1)=2.1192$,
- для 2-й $I=2$ и $V(2,1)=2.6491$,
- для 3-й строки $I=3$ и $V(3,1)=2.1192$.

В столбце также $N=6$ элементов, но остальные равны нулю.

Определив положение и значения всех ненулевых элементов матриц А и В, ввод исходных данных для работы программы 2.1 можно организовать без использования операторов МАТ 35 и 55, в которых элементы матрицы А(I,J) и матрицы-столбца В(I,1) задаются в *полном виде* по строкам. *Применение операторов присваивания позволяет вводить только ненулевые значения элементов соответствующих матриц с указанием их расположения.* При этом нулевые значения элементов можно не указывать, так как все элементы матрицы А(I,J) и матрицы-столбца В(I,1), размерность которых была описана оператором 30, равны 0. Поэтому достаточно только задать в программе 2.1 ненулевые элементы:

A(1,1):=-0.7071; A(1,2):=-1; A(1,3):=-0.6247;	35
A(2,3):=-0.7809; A(3,1):=-0.7071; A(4,1):=0.7071;	40
A(4,6):=0.6172; A(5,5):=-1; A(5,6):=-0.7715;	45
A(6,1):=0.7071; A(6,4):=1; A(6,6):=-0.1543;	50
V(1,1):=2.1192; V(2,1):=2.6491; V(3,1):=2.1192;	55

Напомним, что при описании вариантов и дополнений к основным программам вставляемые операторы, команды или предложения должны располагаться в основной программе по порядку возрастания вспомогательных номеров, при этом:

- *если номер вставляемого оператора совпадает* с имеющимся в основной программе, то последний *заменяется*;
- *если номер не совпадает* — то предложение *вставляется* между операторами с соответствующими меньшим и большим номерами в основной программе.

Дополнение 2.2. Если команду 80 представить в виде

OFF FLOAT, NERO; 80

удалив из программы используемую здесь команду 90, то отмена установленного режима FLOAT для работы с вещественной арифметикой произойдет *до распечатки результатов решения.* Это приведет к тому, что введенные операторами 35 и 45 вещественные числа будут представлены в виде рациональных, о чем для каждого используемого значения числа будет выдано соответствующее сообщение. Результаты решения (2.13) будут также представлены в виде рациональных чисел, отчего их значение, конечно, не изменится:

$$X(1,1) := -\frac{447681}{149375} \qquad X(2,1) := \frac{7001719}{3303891}$$

$$\begin{aligned}
 X(3,1) &:= -\frac{26878}{7923} & X(4,1) &:= \frac{3029791101}{1143748750} & (2.14) \\
 X(5,1) &:= -\frac{2649}{1000} & X(6,1) &:= \frac{5298}{1543}
 \end{aligned}$$

Однако проще не вычислять значения действительных чисел при подготовке исходных данных, а оставить их после определения тригонометрических функций в форме рациональной дроби и корней целых чисел, в которой они естественным образом определяются соотношениями (2.9). Программа 2.1 в этом случае может принять, например, следующий вид:

```

% Программа 2.2
OUT "\C8\PR2-2.LIS";                                10
ON NERO;                                              15
N:=6;                                                 20
SIN(FI):=4/SQRT(41); COS(FI):=5/SQRT(41);           22
SIN(PSI):=4/SQRT(57); COS(PSI):=SQRT(41)/SQRT(57);  24
SIN(BETA):=1/SQRT(2); COS(BETA):=1/SQRT(2);         26
SIN(TETA):=SQRT(41)/SQRT(42); COS(TETA):=1/SQRT(42);  28
P:=4;                                                 29
MATRIX A(N,N), B(N,1), X(N,1);                       30
A:=MAT((-SIN(BETA), -1, -SIN(FI), 0, 0, 0),          35
(0, 0, -COS(FI), 0, 0, 0),
(-COS(BETA), 0, 0, 0, 0, 0),
(SIN(BETA), 0, 0, 0, 0, SIN(TETA)*SIN(FI)),          (2.15)
(0, 0, 0, 0, -1, -SIN(TETA)*COS(FI)),
(COS(BETA), 0, 0, 1, 0, -COS(TETA)));
B:=MAT((P*COS(PSI)*SIN(FI)), (P*COS(PSI)*COS(FI)),    55
(P*SIN(PSI)), (0), (0), (0));
OFF NERO;                                             80
X:=A**(-1)*B;                                        85
SHUT "\C8\PR2-2.LIS";                                95
END;                                                  99

```

Команды и операторы 15, 20, 30, 80, 85 и 99 эквивалентны соответствующим командам и операторам программы 2.1 и повторно не описываются. Отметим отличия программ 2.2 и 2.1.

Команды 10 и 95 отличаются только именем внешнего файла PR2-2.LIS, соответственно открываемого и закрываемого этими командами, куда будут записываться результаты работы программы 2.2.

Команда 15 включает только флаг NERO, чем запрещается печать нулевых значений при контрольной распечатке матриц А и В. Включение режима FLOAT для работы с вещественной арифметикой из команды 15 удалено, что делает ненужным и его выключение командой 90, которая также убрана из программы 2.2 (подробнее см. дополнение 2.4).

Новые операторы присваивания 21–28 выражают в программе 2.2 значения тригонометрических функций в форме рациональной дроби и корней целых чисел, в которой они естественным образом определяются соотношениями (2.9).

Оператор присваивания 29 задает значение силе Р, которая будет использоваться в качестве связанной переменной в общей записи элементов матрицы-столбца В (правых частей уравнений равновесия (2.8)).

Операторы 35 и 55 вводят исходные данные соответственно матрицы А и матрицы-столбца В в символьной форме, элементы которых также задаются в полном виде по строкам сразу из формализованных уравнений (2.8).

Дополнение 2.3. Операторы присваивания 21–28, выражающие тригонометрические функции в программе 2.2, можно также представить в нижеследующей степенной форме:

$$\begin{aligned} \text{SIN}(\text{FI}) &:= 4/41^{1/2}; & \text{COS}(\text{FI}) &:= 5/41^{1/2}; & & 22 \\ \text{SIN}(\text{PSI}) &:= 4/57^{1/2}; & \text{COS}(\text{PSI}) &:= 41^{1/2}/57^{1/2}; & & 24 \\ \text{SIN}(\text{BETA}) &:= 1/2^{1/2}; & \text{COS}(\text{BETA}) &:= 1/2^{1/2}; & & 26 \\ \text{SIN}(\text{TETA}) &:= 41^{1/2}/42^{1/2}; & \text{COS}(\text{TETA}) &:= 1/42^{1/2}; & & 28 \end{aligned}$$

Распечатка результатов решения после окончания работы программы 2.2 при обоих способах задания тригонометрических функций (в программе 2.2 и по дополнению 2.3) будет иметь следующий вид:

$$\begin{aligned} X(1,1) &:= -\frac{16 * \text{sqrt}(2)}{\text{sqrt}(57)} & X(2,1) &:= \frac{16}{\text{sqrt}(57)} \\ X(3,1) &:= -\frac{4 * \text{sqrt}(41)}{\text{sqrt}(57)} & X(4,1) &:= \frac{20}{\text{sqrt}(57)} \end{aligned} \quad (2.16)$$

$$X(5,1) := -\frac{20}{\sqrt{57}} \qquad X(6,1) := \frac{4 * \sqrt{42}}{\sqrt{57}}$$

Конечно, это только точная форма представлений приближенных вещественных значений результатов (2.13) в виде дробей из целых чисел и их корней.

Дополнение 2.4. Если в процессе вычисления встречается действительное (с плавающей точкой) число, то система обычно преобразует его в отношение двух целых чисел и выдает сообщение о проделанном преобразовании.

Если пользователь желает использовать действительную арифметику, он может воспрепятствовать этому преобразованию командой ON FLOAT, которая только запрещает преобразовывать число с плавающей точкой в рациональную дробь во время вычислений.

Поэтому если в команде 15 включить режим FLOAT для работы с вещественной арифметикой (как в программе 2.1)

```
ON FLOAT, NERO;
```

15

то никаких изменений в форме представлений результатов решения (2.16) программы 2.2 не будет, так как среди вводимых данных нет действительных или рациональных чисел.

Кроме этого, CAB REDUCE предназначена для выполнения точных алгебраических преобразований. Поэтому использование приближенных вещественных чисел при сложных вычислениях, к которым относится и определение обратной матрицы при решении СЛАУ, может привести к ошибкам.

Однако использование действительной арифметики удобнее для представления результатов решения в численной форме.

Поэтому все расчеты следует проводить с использованием рациональных чисел, а результаты решения представить в приближенном виде с использованием действительных чисел.

Для этого после окончания расчета в программе 2.2 следует:

- установить флаги BIGFLOAT и NUMVAL командой 87;
- распечатать результаты решения выражением 88, при этом они представят в виде (2.13) с использованием действительных чисел (только вместо идентификатора X будет использоваться MAT);

- отменить установленные на время режимы командой 90:

ON BIGFLOAT, NUMVAL;	87
X;	88
OFF BIGFLOAT, NUMVAL;	90

Вышеописанный результат достигается только при совместном действии флагов BIGFLOAT и NUMVAL, причем вместо флага BIGFLOAT не может использоваться флаг FLOAT.

Флаг BIGFLOAT обеспечивает использование в многочленах вещественных коэффициентов повышенной точности (по умолчанию в этом режиме точность вещественных чисел определяется десятью десятичными числами).

Флаг NUMVAL устанавливает режим вычисления значений элементарных функций. Функции SIN, COS, TAN, ASIN, ACOS, SQRT, EXP, LOG с числовым аргументом и зарезервированные переменные E и PI принимают численные значения в форме с плавающей запятой с текущей степенью точности.

Дополнение 2.5. Ввод исходных данных в программе 2.2 также можно организовать без операторов MAT 35 и 55 с использованием операторов присваивания для *ненулевых значений* элементов соответствующих матриц. В отличие от дополнения 2.1, ненулевые значения элементов здесь указываются с применением тригонометрических функций:

A(1,1) := -SIN(BETA); A(1,2) := -1; A(1,3) := -SIN(FI);	35
A(2,3) := -COS(FI); A(3,1) := -COS(BETA);	40
A(4,1) := SIN(BETA); A(4,6) := SIN(TETA) * SIN(FI);	42
A(5,5) := -1; A(5,6) := -SIN(TETA) * COS(FI);	45
A(6,1) := COS(BETA); A(6,4) := 1; A(6,6) := -COS(TETA);	50
B(1,1) := P * COS(PHI) * SIN(FI);	55
B(2,1) := P * COS(PHI) * COS(FI); B(3,1) := P * SIN(PHI);	60

Так как тригонометрические функции предварительно определены операторами 22–28 (в программе 2.2 и по дополнению 2.3), то при обоих способах их задания распечатка результатов решения (2.16) не изменится.

Программа 2.2 с учетом дополнения 2.5 представляет только другую форму численного решения СЛАУ (2.8) и этим не слишком сильно отличается от программы 2.1. Однако она уже гораздо лучше приспособлена для возможности исследования влияния на результаты решения вариации

различных факторов: величины и направления действующих сил, а также геометрических параметров конструкции.

Дополнение 2.6. В CAB REDUCE можно записать и многократный оператор присваивания, формат которого в общем случае можно представить в следующем виде:

$$\text{выражение} := \text{выражение} := \dots := \text{выражение} \quad (2.17)$$

В нем каждое выражение принимает значение, вычисленное для самого правого выражения.

Отметим, что в любой части оператора (2.17) вместо выражения может использоваться также переменная или элемент массива (см. его простую форму представления (1.12)).

Реализуя такую возможность, операторы 22–28 программы 2.2, задающие значения тригонометрических функций применяемых углов, можно представить в виде, облегчающем запись операторов MAT 35 и 55. Теперь вместо тригонометрических функций просто используются переменные, идентификаторы которых для простоты состоят из первых букв обозначений самой функции и используемого угла:

$$\begin{aligned} SF &:= \sin(FI) := 4/\sqrt{41}; & CF &:= \cos(FI) := 5/\sqrt{41}; & 22 \\ SP &:= \sin(PSI) := 4/\sqrt{57}; & & & 24 \\ CP &:= \cos(PSI) := \sqrt{41}/\sqrt{57}; & & & 25 \\ SB &:= \sin(\text{BETA}) := 1/\sqrt{2}; & CB &:= \cos(\text{BETA}) := 1/\sqrt{2}; & 26 \\ ST &:= \sin(\text{TETA}) := \sqrt{41}/\sqrt{42}; & & & 27 \\ CT &:= \cos(\text{TETA}) := 1/\sqrt{42}; & & & 28 \\ A &:= \text{MAT}((-SB, -1, -SF, 0, 0, 0), & & & 35 \\ & (0, 0, -CF, 0, 0, 0), & & & \\ & (-CB, 0, 0, 0, 0, 0), & & & \\ & (SB, 0, 0, 0, 0, ST*SF), & & & \\ & (0, 0, 0, 0, -1, -ST*CF), & & & \\ & (CB, 0, 0, 1, 0, -CT)); & & & (2.18) \\ B &:= \text{MAT}((P*CP*SF), (P*CP*CF), (P*SP), (0), (0), (0)); & & & 55 \end{aligned}$$

Дополнение 2.7. Фрагмент программы на REDUCE дополнения 2.5, осуществляющий ввод ненулевых значений исходных данных в программе 2.2, также удобнее представить с использованием вспомогательных переменных из многократных операторов присваивания 22–28 дополнения 2.6, что может иметь следующий вид:

$A(1, 1) := -SB$; $A(1, 2) := -1$; $A(1, 3) := -SF$;	35
$A(2, 3) := -CF$; $A(3, 1) := -CB$;	40
$A(4, 1) := SB$; $A(4, 6) := ST * SF$;	42
$A(5, 5) := -1$; $A(5, 6) := -ST * CF$;	45
$A(6, 1) := CB$; $A(6, 4) := 1$; $A(6, 6) := -CT$;	50
$B(1, 1) := P * CP * SF$;	55
$B(2, 1) := P * CP * CF$; $B(3, 1) := P * SP$;	60

Дополнение 2.8. В системе REDUCE в простых случаях нет необходимости в команде вывода информации на терминал, поскольку значение любого выражения печатается автоматически при использовании ограничителя ";". Однако в REDUCE для этой же цели введена специальная команда WRITE, которая снимает некоторые ограничения, связанные с использованием ограничителя ";" в некоторых сложных случаях. Формат команды следующий:

WRITE список параметров, разделенных запятыми;

В качестве параметров могут использоваться:

- *выражения* (включая переменные и константы), которые при выводе оцениваются и печатаются;
- *операторы присваивания*, в которых выражение, стоящее в правой части, при выводе оценивается, операция присваивания выполняется и распечатывается вместе с левой частью;
- *произвольный текст, заключённый в кавычки*, выводится в том же виде без кавычек на печать.

Все значения указанных параметров вместе с сопровождающим текстом печатаются одна за другой на данной строке. Для их отделения друг от друга в кавычках указываются пробелы.

Если количество выводимой информации превышает длину строки, то печать продолжается на другой строке. После оценивания команды WRITE печатаемая строка закрывается. Поэтому список параметров, состоящий только из одного пробела в кавычках (WRITE " ");, приведёт к пропуску пустой строки.

Программу 2.2 желательно дополнить поясняющими сообщениями, что можно сделать, например, следующим образом:

WRITE "ВХОДНАЯ МАТРИЦА ЛЕВОЙ ЧАСТИ A";	33
WRITE "ВЕКТОР ПРАВЫХ ЧАСТЕЙ B";	53

WRITE "РЕЗУЛЬТАТЫ РЕШЕНИЯ СИСТЕМЫ УРАВНЕНИЙ"; 68

В данном случае поясняющие сообщения команды WRITE можно заменить использованием комментария, например, для команды 33:

COMMENT ВХОДНАЯ МАТРИЦА ЛЕВОЙ ЧАСТИ А; 33

или

% ВХОДНАЯ МАТРИЦА ЛЕВОЙ ЧАСТИ А 33

Отметим, что для выполнения WRITE, как и для любой команды, безразлично, какой после нее использован ограничитель ";" или "\$": в любом случае она *выполняется* и указанный список распечатывается.

2.2.3 Решение в символьном виде. Исследование влияния вариации нагрузки

Представляется исключительно интересным исследовать влияние параметров СЛАУ на результаты решения. Применительно к задачам статики это означает исследование влияния вариации нагрузки и геометрических факторов на значения реакций опор или усилий в стержнях конструкций. При численном решении на ПК с использованием алгоритмических языков для этого нужно проделать большую работу по подготовке исходных данных [3, 17.]

CAB REDUCE поразительно легко справляется с подобной задачей. Это достигается использованием в процессе символьных вычислений *свободных переменных и свободных имен со скобками*.

Основанием для этого служит тот факт, что в REDUCE различают имя переменной и ее значение, даже если они совпадают. Каждая переменная обозначается именем, которое является ее первоначальным значением.

Имя переменной — обозначение “емкости”, которую с помощью операторов присваивания наполняют содержимым (численным или символьным значением).

Свободной называется переменная, которой ничего не присвоено и значение которой совпадает с ее именем.

Связанной — которой ранее было присвоено некоторое значение.

Для свободных переменных алгебраические преобразования выполняются с их именами, которые и входят в окончательный ответ, определяя его зависимость от значения последних.

Всем использованным переменным ранее всегда предварительно давались значения, то есть они являлись связанными. Поэтому если из программы 2.2 удалить оператор 29, сделав свободной переменной величину нагрузки силу P , очистив для надежности ее значение командой

29

то мы получим значения усилий в стержнях рассматриваемой пространственной конструкции в зависимости от значения силы P :

$$\begin{aligned}
 X(1,1) &:= -\frac{4 * \text{sqrt}(2) * p}{\text{sqrt}(57)} & X(2,1) &:= \frac{4 * p}{\text{sqrt}(57)} \\
 X(3,1) &:= -\frac{\text{sqrt}(41) * p}{\text{sqrt}(57)} & X(4,1) &:= \frac{5 * p}{\text{sqrt}(57)} \\
 X(5,1) &:= -\frac{5 * p}{\text{sqrt}(57)} & X(6,1) &:= \frac{\text{sqrt}(42) * p}{\text{sqrt}(57)}
 \end{aligned} \tag{2.19}$$

Их графики представляют собой прямые линии, выходящие из начала координат, с тангенсами углов наклона, равными коэффициентам перед силой P в соотношениях (2.19).

Еще более впечатляющий результат достигается использованием свободных имен со скобками. В REDUCE основным свойством оператора является то, что само его имя вместе со списком параметров, может быть его значением. Иными словами, *имя со скобкой, при использовании его в качестве оператора может условно рассматриваться в качестве свободной переменной.*

Это свойство свободных имен со скобками при использовании в их качестве системных функций, являющихся также разновидностью операторов, позволяет нам с легкостью проводить исследование вариаций различных геометрических факторов и направлений действующих нагрузок в различных задачах.

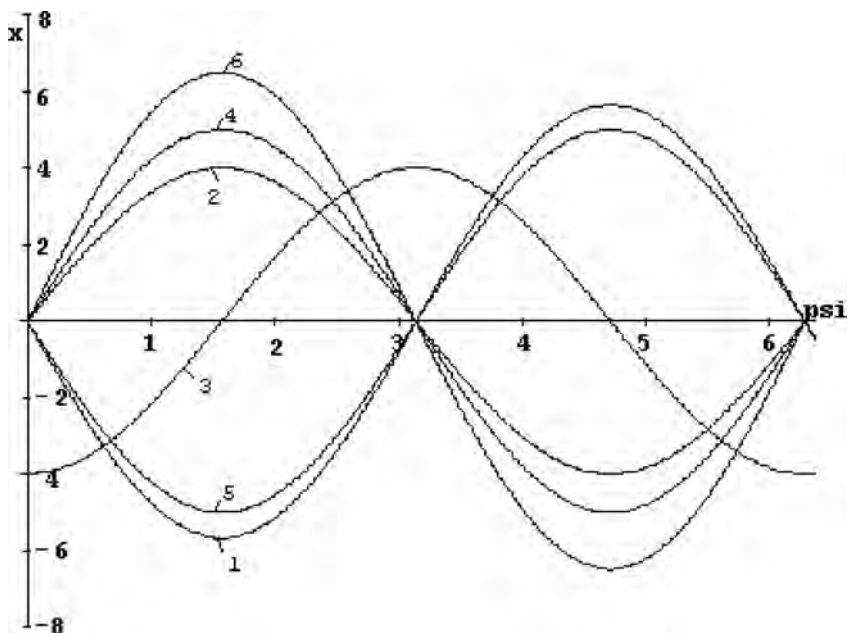


Рис. 2.2. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения угла PSI (рад) при вращении силы P в диагональной плоскости: 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

В программе 2.2 операторы 24 задают значения $\text{SIN}(\text{PSI})$ и $\text{COS}(\text{PSI})$, которые определяют направление действия силы P в диагональной плоскости. Если восстановить в программе 2.2 оператор 29, а удалить операторы 24, то имена со скобками $\text{SIN}(\text{PSI})$ и $\text{COS}(\text{PSI})$, идентифицирующие тригонометрические функции, можно условно рассматривать в качестве свободных переменных. В результате работы такой измененной программы мы получим значения усилий в стержнях пространственной конструкции в зависимости от вариации направления постоянной по модулю силы P в диагональной плоскости:

$$\begin{aligned}
 X(1,1) &:= -4 * \text{sqrt}(2) * \sin(\text{psi}) & X(2,1) &:= 4 * \sin(\text{psi}) \\
 X(3,1) &:= -4 * \cos(\text{psi}) & X(4,1) &:= 5 * \sin(\text{psi}) \\
 X(5,1) &:= -5 * \sin(\text{psi}) & X(6,1) &:= \text{sqrt}(42) * \sin(\text{psi})
 \end{aligned}
 \tag{2.20}$$

Графики на рис. 2.2 наглядно показывают зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения угла Ψ (рад) при вращении силы P в диагональной плоскости. Как видно из их сравнения, значения усилий в стержнях пространственной конструкции при вращении постоянной по модулю силы P в диагональной плоскости изменяются по синусоидальному закону:

В программе 2.2 операторы 22 задают значения $\text{SIN}(FI)$ и $\text{COS}(FI)$, которые определяют направление действия силы P при ее вращении вокруг вертикальной оси с постоянным углом Ψ . Если восстановить в программе 2.2 операторы 24, а удалить операторы 22, то в результате ее работы мы получим зависимости результатов решения от изменения направления постоянной по модулю силы P при ее вращении вокруг вертикальной оси с постоянным углом наклона к горизонтальной плоскости:

$$\begin{aligned} X(1,1) &:= -\frac{16 * \text{sqrt}(2)}{\text{sqrt}(57)} & X(2,1) &:= \frac{16}{\text{sqrt}(57)} \\ X(3,1) &:= -\frac{4 * \text{sqrt}(41)}{\text{sqrt}(57)} & X(4,1) &:= \frac{16 * (\text{sqrt}(41) * \sin(fi) + 1)}{\text{sqrt}(2337) * \sin(fi)} \quad (2.21) \\ X(5,1) &:= -\frac{16 * \cos(fi)}{\text{sqrt}(57) * \sin(fi)} & X(6,1) &:= \frac{16 * \text{sqrt}(42)}{\text{sqrt}(2337) * \sin(fi)} \end{aligned}$$

Простота выполнения вариаций различных факторов открывает такие широкие возможности, которые могут просто погубить легкомысленного исследователя, о чем в аналогичной ситуации предупреждал еще Бармалея всем известный добрый доктор в фильме «Айболит-66».

Применительно к нам это означает, что начав исследовать влияние вращения силы P путем вариации угла φ (идентификатором которого в уравнениях 2.21 является FI), мы не обратили внимание, что он также определяет положение третьего стержня конструкции (см. рис. 53, [22, с. 53]), являясь углом между диагоналями верхней и нижней плоскости и горизонтальной прямой.

Поэтому при вращении силы P мы также *несогласованным образом* стали изменять размеры конструкции. При $\varphi = 0$ или $\varphi = \pi$ *пространственная шарнирно-стержневая конструкция вырождается в плоскую*, которая не может существовать при заданных размерах.

В результате геометрической несогласованности происходит потеря равновесия и устойчивости конструкции, что математически находит свое отражение в стремлении к бесконечности реакций трех последних стержней 4–6 (2.21), где в выражениях для $X(4,1) - X(6,1)$ при $\varphi = 0$ или $\varphi = \pi$ происходит деление на 0.

Это очень хорошо видно на графиках зависимости значений реакций стержней 1–6 от величины угла φ , построенных по соотношениям (2.21):

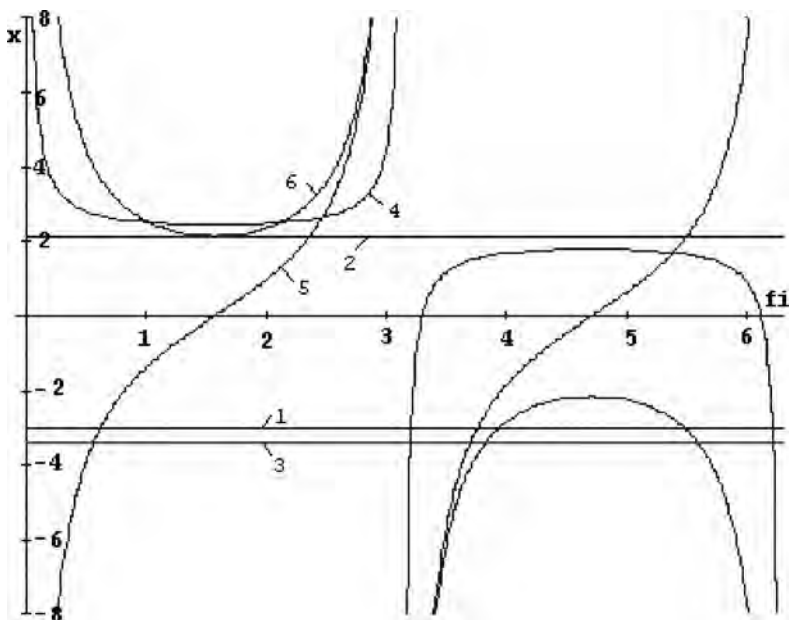


Рис. 2.3. Зависимости усилий (X , кН) в стержнях 1-6 пространственной фермы при формальной вариации угла φ (рад) для всей конструкции (рис. 2.1): 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

Чтобы этого не происходило, нужно угол между вертикальной плоскостью действия силы P и горизонтальной осью обозначить другой буквой, например α (с идентификатором ALFA).

Его значение для начального положения равно φ , но при вращении силы P вокруг вертикальной оси угол α изменяется соответствующим образом независимо от постоянного значения угла φ . В программе 2.2

оператор 55 с учетом этого примет следующий вид (где жирным шрифтом выделены измененные обозначения):

$$B := \text{MAT} \left((P * \cos(\text{PSI}) * \sin(\mathbf{ALFA})), (P * \cos(\text{PSI}) * \cos(\mathbf{ALFA})), \quad 55 \right. \\ \left. (P * \sin(\text{PSI})), (0), (0), (0) \right); \quad (2.22)$$

Значение угла ALFA в программе 2.2 не задано, поэтому он является свободной переменной.

Теперь в результате работы такой измененной программы 2.2 с новым оператором 55 мы получим настоящие зависимости результатов решения (2.23) от вариации угла α при вращении силы P вокруг вертикальной оси с постоянным углом наклона к горизонтальной плоскости. Они весьма сильно отличаются от легкомысленно полученных соотношений (2.21), что видно из их сравнения:

$$X(1,1) := - \frac{16 * \text{sqrt}(2)}{\text{sqrt}(57)}$$

$$X(2,1) := \frac{4 * (4 * \text{sqrt}(41) * \cos(\text{alfa}) - 5 * \text{sqrt}(41) * \sin(\text{alfa}) + 20)}{5 * \text{sqrt}(57)}$$

$$X(3,1) := - \frac{164 * \cos(\text{alfa})}{5 * \text{sqrt}(57)} \quad X(4,1) := \frac{20}{\text{sqrt}(57)} \quad (2.23)$$

$$X(5,1) := - \frac{20}{\text{sqrt}(57)} \quad X(6,1) := \frac{4 * \text{sqrt}(42)}{\text{sqrt}(57)}$$

Графики зависимостей значений реакций стержней 1–6 пространственной конструкции (рис. 2.1) от изменения угла ALFA (рад) при вращении силы P вокруг вертикальной оси, построенных по соотношениям (2.23) на рис. 2.4, также наглядно подтверждают это.

Оказывается, значение реакций при вращении силы P вокруг вертикальной оси с постоянным углом наклона к горизонтальной плоскости ψ :

- в стержнях 1, 4–6 является постоянным,
- в третьем — изменяется по косинусоидальному закону,
- для второго стержня — определяется разностью функций $\cos\alpha$ и $\sin\alpha$, а сама зависимость смещена по вертикальной оси от нулевого значения.

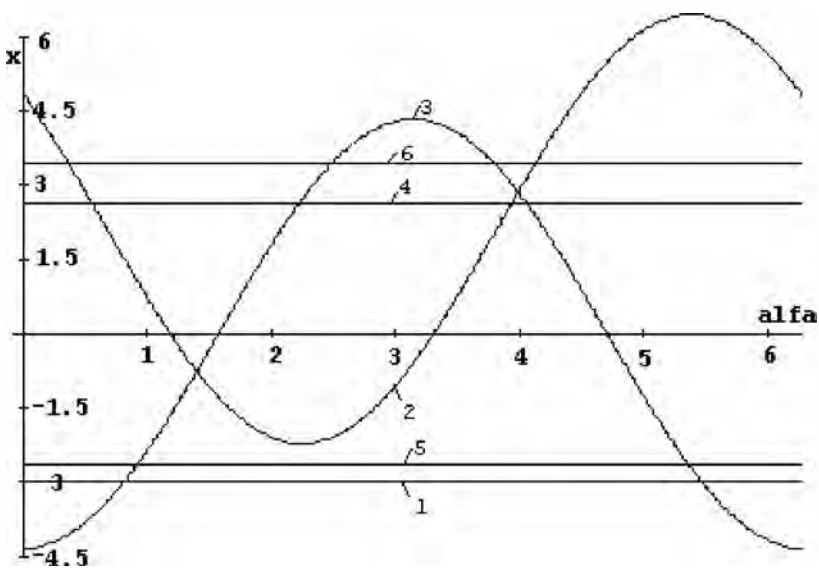


Рис. 2.4. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения угла $ALFA$ (рад) при вращении силы P вокруг вертикальной оси: 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

Одновременное удаление из программы 2.2 операторов 24, 29 при изменении оператора 55 по фрагменту (2.22) даст возможность получить зависимости значений реакций пространственной конструкции (рис. 2.1) при всевозможном изменении величины и направления силы P :

$$X(1,1) = -\sqrt{2} \cdot \sin(\psi) \cdot p$$

$$X(2,1) = (p \cdot (4 \cdot \cos(\alpha) \cdot \cos(\psi) - 5 \cdot \cos(\psi) \cdot \sin(\alpha) + 5 \cdot \sin(\psi))) / 5$$

$$X(3,1) = -\frac{\sqrt{41} \cdot \cos(\alpha) \cdot \cos(\psi) \cdot p}{5} \quad (2.24)$$

$$X(4,1) = \frac{5 \cdot \sin(\psi) \cdot p}{4} \quad X(5,1) = -\frac{5 \cdot \sin(\psi) \cdot p}{4}$$

$$X(6,1) = \frac{\sqrt{42} \cdot \sin(\psi) \cdot p}{4}$$

Теперь слишком широкие возможности при выполнении вариаций нагрузки просто придавливают нас тяжестью последующего анализа зависимостей реакций в стержнях от двух-трех переменных параметров (P , φ и α) в соотношениях (2.24).

Поэтому здесь, как и во многих других случаях, можно лишний раз вспомнить, как там у Епифания Премудрого сказано: *“Простота без простоты”* и *“Не мудрствуя лукаво”*. Затем, будучи ободренными этими древними методическими принципами, в дальнейшем всегда будем проводить исследования:

- варьируя только один переменный параметр по образцам (2.19), (2.20) и (2.22),
- дополняя полученные аналитические соотношения их графическим двумерным представлением типа рис. 2.2 – 2.4,
- внимательно осмысливая и анализируя полученные аналитические и графические зависимости, определяя зону оптимальной действующей нагрузки и других варьируемых параметров. Ведь этого за вас не сможет сделать ПК! И теперь, как и ранее, остается верным принцип: *“Если не умеешь думать, нечего садиться за компьютер!”*.

2.2.4 Исследование влияния вариации геометрических факторов

При вариации геометрических факторов нас еще больше поражают открывающиеся перед нами возможности, однако, и опасности возрастают. Ведь недаром расчеты на ПК сравнивают с прогулкой в горах: чем выше забираешься, тем большие красоты открываются твоему взору, но и возможности свалиться в пропасть увеличиваются.

Поэтому, утвердившись в мысли, что *“лучше гор могут быть только”* ... *символьные аналитические преобразования на компьютере*, вернемся к теме нашего исследования.

Основной опасностью при вариации геометрических факторов являются несогласованные изменения размеров конструкции.

Ранее, начав исследовать влияние вращения силы P путем вариации угла φ при получении уравнений 2.21, мы обратили на это внимание. Рассмотрим этот вопрос подробнее.

При вариации угла φ происходит изменение его тригонометрических функций (первые два уравнения соотношений (2.9)), которые определяются размерами **a** и **b**. Это приводит к соответствующему *изменению этих величин*. Но они входят в свою очередь в определение остальных тригонометрических функций (2.9): **a** для трех (ψ, β, θ) и **b** для двух (ψ, θ), *которые остались постоянными*. Значит, в этих соотношениях **a** и **b** не изменились.

Это противоречие и называется *несогласованными изменениями размеров конструкции*, при которых она в данном виде не может существовать.

Значит, потеря равновесия и устойчивости данной конструкции происходит не только в особых точках при $\varphi = 0$ или $\varphi = \pi$, в которых в уравнениях (2.21) происходит деление на 0, а этим соотношениям нельзя доверять при любых значениях угла φ , так как они не имеют никакого отношения к рассматриваемой конструкции.

Поэтому в программе 2.2 нельзя удалить операторы 22–29 и, получив соотношения (2.25) считать, что они описывают значения реакций стержней при всевозможных согласованных изменениях направлений стержней в плоскостях их действия и силы P . Это будет просто результат неосознанного вождения самого себя за нос:

$$\begin{aligned}
 X(1,1) &:= - \frac{\sin(\psi) * p}{\cos(\beta)} & X(2,1) &:= \frac{\sin(\beta) * \sin(\psi) * p}{\cos(\beta)} \\
 X(3,1) &:= - \cos(\psi) * p & & (2.25) \\
 X(4,1) &:= \frac{(\sin(\psi) * p * (\cos(\beta) * \sin(\varphi) * \sin(\theta) + \cos(\theta) * \sin(\beta)))}{(\cos(\beta) * \sin(\varphi) * \sin(\theta))} \\
 X(5,1) &:= - \frac{\cos(\varphi) * \sin(\beta) * \sin(\psi) * p}{\cos(\beta) * \sin(\varphi)} \\
 X(6,1) &:= \frac{\sin(\beta) * \sin(\psi) * p}{\cos(\beta) * \sin(\varphi) * \sin(\theta)}
 \end{aligned}$$

На самом деле для выполнения исследования влияния вариации геометрических факторов на значения реакций стержней нужно:

- в программе 2.2 представить выражения для тригонометрических функций в общем виде через размеры данной конструкции **a**, **b**, **c** и **d**, например, с использованием идентификаторов AR, BR (для отличия от соответствующих матриц A, B), C и D:

$$\text{SIN (FI) :=AR/SQRT (AR**2+BR**2) ;} \quad 21$$

$$\text{COS (FI) :=BR/SQRT (AR**2+BR**2) ;} \quad 22$$

$$\text{SIN (PSI) :=C/SQRT (AR**2+BR**2+C**2) ;} \quad 23$$

$$\text{COS (PSI) :=SQRT (AR**2+BR**2) /SQRT (AR**2+BR**2+C**2) ;} \quad 24$$

$$\text{SIN (BETA) :=AR/SQRT (AR**2+C**2) ;} \quad 25$$

$$\text{COS (BETA) :=C/SQRT (AR**2+C**2) ;} \quad 26$$

$$\text{SIN (TETA) :=SQRT (AR**2+BR**2) /SQRT (AR**2+BR**2+D**2) ;} \quad 27$$

$$\text{COS (TETA) :=D/SQRT (AR**2+BR**2+D**2) ;} \quad 28$$

- предварительно операторами присваивания следует задать исходные размеры данной конструкции:

$$\text{AR:=4; BR:=5; C:=4; D:=1;} \quad 18$$

- представить оператор 55 в одной из двух форм с использованием:

- *численных значений*, задающих имеющиеся в нем тригонометрические функции, для исследования влияния вариации геометрических факторов на значения реакции стержней:

$$\begin{aligned} \text{B:=MAT ((P*SQRT (41/57) *4/SQRT (41)) ,} & \quad 55 \\ \text{(P*SQRT (41/57) *5/SQRT (41)) ,} & \quad (2.26) \\ \text{(P*4/SQRT (57)) , (0) , (0) , (0)) ;} & \end{aligned}$$

- *символьных обозначений* для возможности независимого варьирования направления действия силы P, применяя для задания имеющихся в нем углов *свободные переменные* GAMMA и ALFA, не имеющие никаких значений в программе 2.2 и не используемые там более нигде, например:

$$\begin{aligned} \text{B:=MAT ((P*COS (GAMMA) *SIN (ALFA)) ,} & \quad 55 \\ \text{(P*COS (GAMMA) *COS (ALFA)) ,} & \quad (2.27) \\ \text{(P*SIN (GAMMA)) , (0) , (0) , (0)) ;} & \end{aligned}$$

Теперь достаточно осознанно скомпоновать универсальную программу 2.3, приспособленную для вариации геометрических размеров конструкции.

Это можно сделать, например, следующим образом:

% Программа 2.3

```

OUT "\C8\PR2-3.LIS"; 10
ON NERO; 15
AR:=4; BR:=5; C:=4; D:=1; 18
N:=6; 20
SIN(FI):=AR/SQRT(AR**2+BR**2); 21
COS(FI):=BR/SQRT(AR**2+BR**2); 22
SIN(PSI):=C/SQRT(AR**2+BR**2+C**2); 23
COS(PSI):=SQRT(AR**2+BR**2)/SQRT(AR**2+BR**2+C**2); 24
SIN(BETA):=AR/SQRT(AR**2+C**2); 25
COS(BETA):=C/SQRT(AR**2+C**2); 26
SIN(TETA):=SQRT(AR**2+BR**2)/SQRT(AR**2+BR**2+D**2); 27
COS(TETA):=D/SQRT(AR**2+BR**2+D**2); 28
P:=4; 29
MATRIX A(N,N), B(N,1), X(N,1); 30
A:=MAT((-SIN(BETA), -1, -SIN(FI), 0, 0, 0), 35
(0, 0, -COS(FI), 0, 0, 0),
(-COS(BETA), 0, 0, 0, 0, 0), (2.28)
(SIN(BETA), 0, 0, 0, 0, SIN(TETA)*SIN(FI)),
(0, 0, 0, 0, -1, -SIN(TETA)*COS(FI)),
(COS(BETA), 0, 0, 1, 0, -COS(TETA)));
B:=MAT((P*SQRT(41/57)*4/SQRT(41)), 55
(P*SQRT(41/57)*5/SQRT(41)),
(P*4/SQRT(57)), (0), (0), (0));
OFF NERO; 80
X:=A**(-1)*B; 85
SHUT "\C8\PR2-3.LIS"; 95
END; 99

```

Все команды и операторы программы 2.3 рассматривались ранее и повторно не описываются. Отметим, что это тестовый вариант программы 2.3, в результате работы которой должны получиться результаты решения в форме (2.16), в чем следует убедиться, запустив ее на выполнение. *Так следует всегда поступать: перед началом вариации удостовериться на известном численном примере в правильности результата базовой программы*, подготовленной для проведения исследований.

Для изучения раздельного влияния вариации геометрических факторов на значения реакций стержней нужно из предложения 18 исключить соответствующий оператор присваивания, задающий данный размер a , b ,

с или **d**. *Предварительно нужно очистить исключаемую переменную, указав ее в качестве параметра команды CLEAR и сделав свободной.*

Если этого не осуществить, то после запуска программы 2.3 *в памяти системы в течение сеанса работы останутся все заданные там переменные.*

Поэтому нельзя будет запускать без выхода из CAB REDUCE предварительно подготовленные файлы с модификациями предложения 18 программы 2.3 или даже с его полным отсутствием. В этом случае никакие исключения из соответствующих операторов присваивания, задающих данный размер конструкции **a**, **b**, **c** или **d**, не будут давать никаких изменений в форме результатов решения (2.16). Система будет помнить в течение сеанса работы все заданные ей переменные и брать недостающие данные из своей памяти.

Поэтому перед выполнением каких-либо вариаций всегда следует очищать исключаемую переменную. Будем считать это признаком хорошего тона, и всегда использовать для этой цели команду **CLEAR**.

В результате предложение 18 программы 2.3 для изучения раздельного влияния вариации геометрических факторов на значения реакций стержней должно принимать следующие формы (2.29), (2.31), (2.33), (2.35). Теперь соответствующий идентификатор AR, BR, C и D становится свободной переменной и входит в символьном виде в результаты решения, выражая их зависимость от данной величины. При этом *все изменения размеров конструкции происходят согласованно*, а результаты решения легко поддаются аналитическому и графическому анализу по аналогии с вышерассмотренной вариацией силы P, что мы предлагаем выполнить самостоятельно для ваших вариантов заданий.

Приведем для рассматриваемого типового примера пространственной шарнирно-стержневой конструкции [22, с. 52-54] формы предложения 18 программы 2.3 с полученными соответствующими аналитическими зависимостями, выражающими влияние раздельного изменения размеров конструкции **a**, **b**, **c** и **d** на значения реакций стержней. Покажем также геометрическую интерпретацию полученных соотношений. В результате все это примет соответственно следующий вид:

- вариация размера **a**, обозначаемого идентификатором AR (в распечатке результатов ar):

$$\text{CLEAR AR; BR:=5; C:=4; D:=1;} \quad (2.29)$$

$$X(1,1) := -\frac{4 * \sqrt{ar^2 + 16}}{\sqrt{57}}$$

$$X(2,1) := \frac{8 * (ar - 2)}{\sqrt{57}}$$

$$X(3,1) := -\frac{4 * \sqrt{ar^2 + 25}}{\sqrt{57}}$$

$$X(4,1) := \frac{20}{\sqrt{57}} \quad (2.30)$$

$$X(5,1) := -\frac{20}{\sqrt{57}}$$

$$X(6,1) := \frac{4 * \sqrt{ar^2 + 26}}{\sqrt{57}}$$

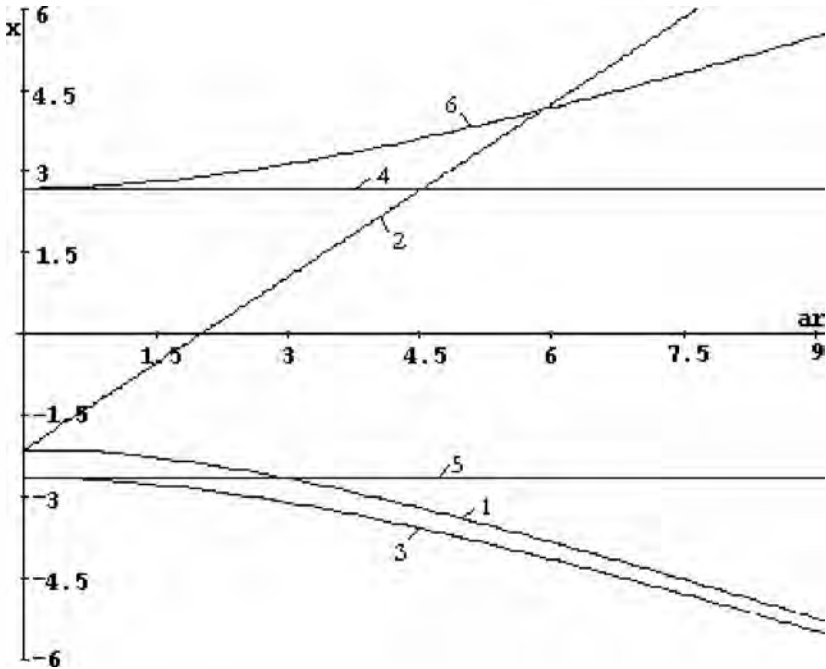


Рис. 2.5. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения ее размера a (м): 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

- вариация размера b , обозначаемого идентификатором BR (в распечатке результатов br):

$$\text{CLEAR BR; AR:=4; C:=4; D:=1;} \quad (2.31)$$

$$X(1,1) := -\frac{16 * \text{sqrt}(2)}{\text{sqrt}(57)}$$

$$X(2,1) := \frac{80}{\text{sqrt}(57) * br}$$

$$X(3,1) := -\frac{20 * \text{sqrt}(br^2 + 16)}{\text{sqrt}(57) * br}$$

$$X(4,1) := \frac{20}{\text{sqrt}(57)} \quad (2.32)$$

$$X(5,1) := -\frac{4 * br}{\text{sqrt}(57)}$$

$$X(6,1) := \frac{4 * \text{sqrt}(br^2 + 17)}{\text{sqrt}(57)}$$

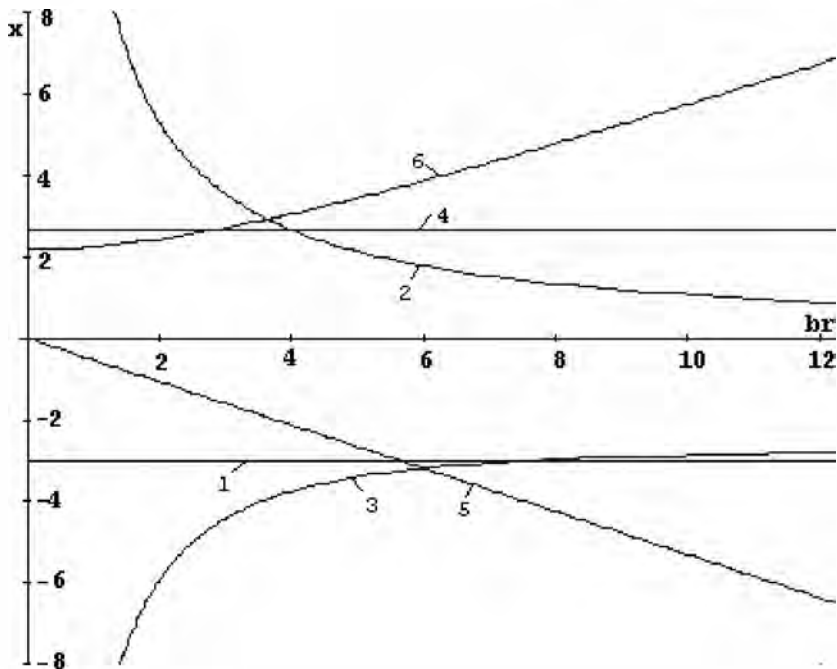


Рис. 2.6. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения ее размера b (м): 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

- вариация размера c , обозначаемого идентификатором C (в распечатке результатов c):

$$\text{CLEAR } C; \text{ AR}:=4; \text{ BR}:=5; \text{ D}:=1; \quad (2.33)$$

$$X(1,1) := - \frac{16 * \text{sqrt}(c^2 + 16)}{\text{sqrt}(57) * c}$$

$$X(2,1) := \frac{64}{\text{sqrt}(57) * c}$$

$$X(3,1) := - \frac{4 * \text{sqrt}(41)}{\text{sqrt}(57)}$$

$$X(4,1) := \frac{16 * (c + 1)}{\text{sqrt}(57) * c} \quad (2.34)$$

$$X(5,1) := - \frac{80}{\text{sqrt}(57) * c}$$

$$X(6,1) := \frac{16 * \text{sqrt}(42)}{\text{sqrt}(57) * c}$$

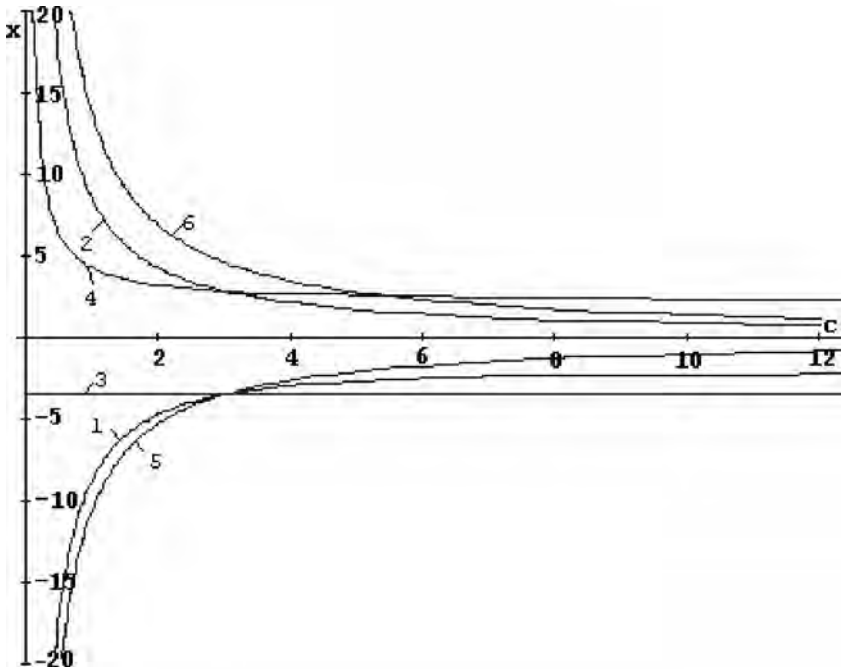


Рис. 2.7. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения ее размера c (м): 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

- вариация размера d , обозначаемого идентификатором D (в распечатке результатов d):

$$\text{CLEAR } D; \text{ AR}:=4; \text{ BR}:=5; \text{ C}:=4; \quad (2.35)$$

$$X(1,1) := -\frac{16 * \text{sqrt}(2)}{\text{sqrt}(57)}$$

$$X(2,1) := \frac{16}{\text{sqrt}(57)}$$

$$X(3,1) := -\frac{4 * \text{sqrt}(41)}{\text{sqrt}(57)}$$

$$X(4,1) := \frac{4 * (d + 4)}{\text{sqrt}(57)} \quad (2.36)$$

$$X(5,1) := -\frac{20}{\text{sqrt}(57)}$$

$$X(6,1) := \frac{4 * \text{sqrt}(d^2 + 41)}{\text{sqrt}(57)}$$

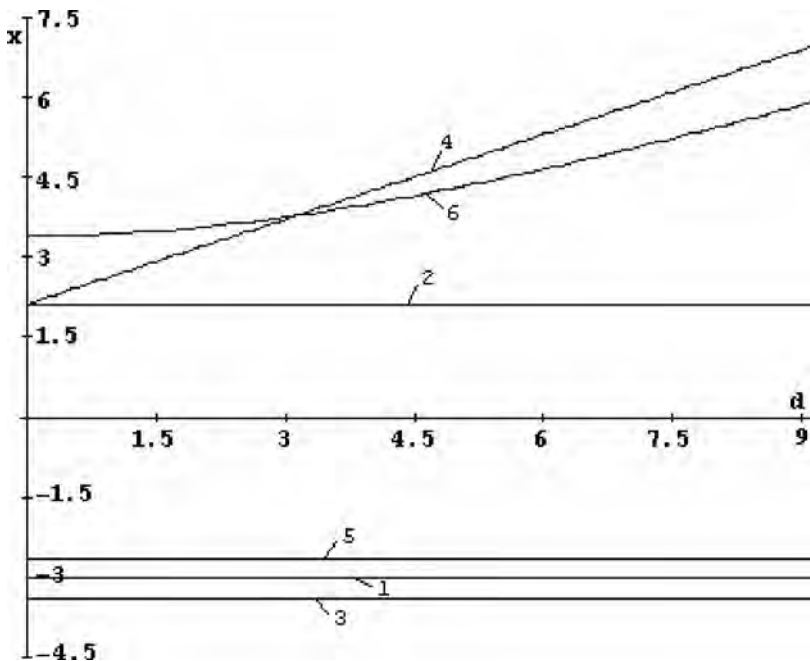


Рис. 2.8. Зависимости усилий (X , кН) в стержнях 1-6 пространственной конструкции (рис. 2.1) от изменения ее размера d (м):
 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6

Полностью очистим все используемые переменные и имена со скобками командой (2.37):

$$\text{CLEAR AR, BR, C, D, P, SIN(ALFA), COS(ALFA), SIN(GAMMA), COS(GAMMA);} \quad (2.37)$$

удалим из программы 2.3 предложение 18 и оператор 29, а также представим оператор 55 по фрагменту (2.27). Теперь после работы такой модифицированной программы 2.3 получим соотношения (2.38), описывающие значения реакций при всевозможных изменениях величины и направления силы P и направлений наклонных стержней в плоскостях их действия:

$$\begin{aligned} X(1,1) &:= -\sqrt{a^2 + c^2} * \sin(\gamma) * p / c \\ X(2,1) &:= (p * (\cos(\alpha) * \cos(\gamma) * a * c - \cos(\gamma) * \sin(\alpha) * b * c + \\ &\quad \sin(\gamma) * a * b)) / (b * c) \\ X(3,1) &:= -\sqrt{a^2 + b^2} * \cos(\alpha) * \cos(\gamma) * p / b \quad (2.38) \\ X(4,1) &:= \sin(\gamma) * p * (c + d) / c \\ X(5,1) &:= -\sin(\gamma) * b * p / c \\ X(6,1) &:= \sqrt{a^2 + b^2 + d^2} * \sin(\gamma) * p / c \end{aligned}$$

Выражения такого общего вида позволяют легко получать значения реакций при любом наборе данных для всей конструкции и действующей нагрузки, проверяя наши предположения, сделанные на основании анализа вариаций каких-либо отдельных факторов.

Как видим, они сильно отличаются от механически полученных ранее уравнений (2.25), наглядно показывающих, как сильно можно увести себя не в ту сторону при бездумном выполнении подобных исследований.

Замечание. *Сложные аналитические преобразования до недавнего времени были доступны только посвященным. Всех остальных они приводили в священный трепет, как что-то им совсем недоступное. Хочется решительно предостеречь от подобного чувства испуга при взгляде на все приведенные аналитические выражения и их графическую интерпретацию. Все они получены ПК в результате ваших очень простых, но аккуратных и внимательных действий, полностью описанных в настоящем пособии.*

Настало время, когда рутинную аналитическую работу практически любой степени сложности с легкостью берет на себя персональный компьютер. Для человека она сопряжена с большими техническими трудностями, в результате которых возрастает вероятность ошибок при действиях “вручную”. В некоторых случаях результат практически невозможно получить без использования ПК.

Однако это не означает, что теперь не нужно будет учить математику, мучиться над анализом результатов, осмысливать их, делая выводы, а все за вас сделает компьютер.

Весьма вероятно, что вами будет выполнена примерно такая же *умственная работа*. При использовании ПК *она просто перераспределяется: ее рутинная часть резко уменьшается, но зато также должна возрасти ее творческая составляющая* по осмыслению и анализу целого спектра всевозможных аналитических решений практически любой степени сложности. Вследствие чего и результаты также резко возрастут. Если этого не произойдет, вы будете просто с использованием современных информационных технологий элегантно водить себя за нос. Но это удовольствие прекратится и идиллия сразу нарушится при первом же практическом использовании результатов вашего анализа, что закончится весьма печально для вас (и зачастую, к большому сожалению, для окружающих).

Это замечание в особой степени касается проведения исследований влияния вариации различных факторов для выбора оптимальных размеров конструкции или условий нагружения.

2.3 РЕШЕНИЕ СЛАУ С “БОЛЬШИМ” КОЛИЧЕСТВОМ УРАВНЕНИЙ

2.3.1 Формализация задачи

Алгоритм решения различных задач статики с использованием САВ REDUCE одинаков, поэтому их программные реализации аналогичны. Это обуславливается тем фактором, что любая задача статики по определению реакций опор (тела или системы тел) или усилий в стержнях шарнирно-стержневой конструкции представляет из себя систему линейных алгебраических уравнений.

Таким образом, любые конкретные реализации пространственных или плоских объектов равновесия или их систем *различаются только количеством систем уравнений, размерами получающихся матриц и значениями их элементов*. Поэтому после достаточно детального рассмотрения п. 2.2 можно было переходить к дальнейшему изучению САВ REDUCE.

Однако столь длительное время для различных разделов статики писались разные программы вплоть до недавнего времени [21], что лучше доказать эти общие рассуждения на конкретных примерах.

В качестве примера задачи статики с “большим” количеством уравнений равновесия воспользуемся аналитическим решением типового задания по определению усилий в стержнях плоской шарнирно-стержневой конструкции для ССС [22, с. 5-12].

Его расчетная схема приведена на рис. 2.9а.

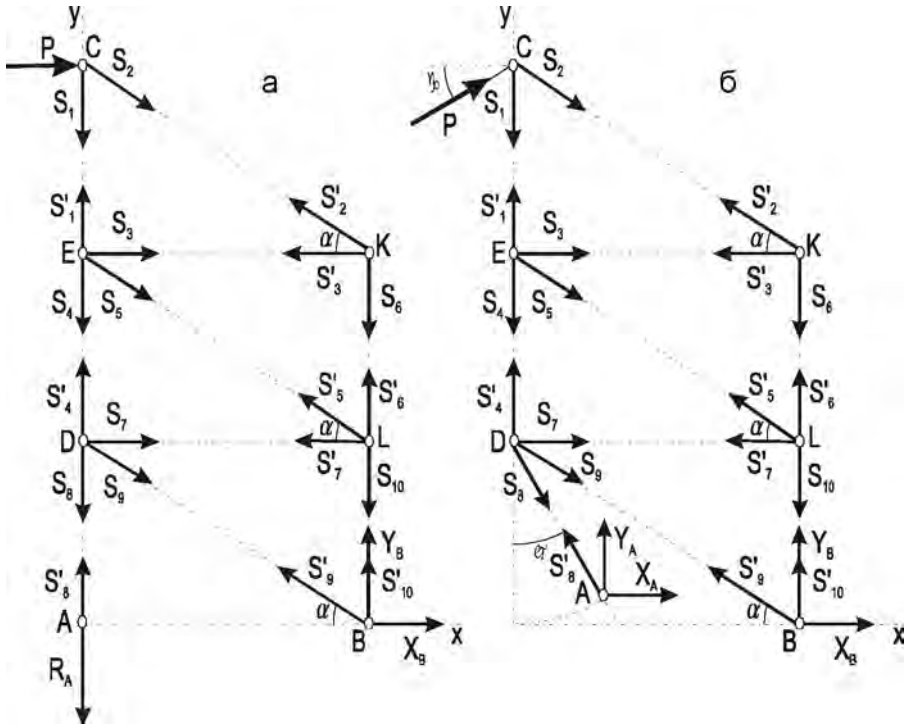


Рис. 2.9. Расчетная схема плоской фермы: а — исходное положение; б — вариации направления силы P и положения шарнирно-неподвижной опоры в точке A

Отличие рис. 2.9, а от рис.5 [22, с. 10] состоит только в том, что реакция R_b показана в общем виде в форме проекций X_b и Y_b. Это обусловлено тем фактом, что при решении СЛАУ задач статики с “большим” количеством уравнений на ПК *не требуется предварительного определения реакций опор*, с которого обычно начинается аналитическое решение. Это делают только для облегчения последующих “ручных” преобра-

зований. Поэтому на рис.5 [22, с. 10] показано *истинное* направление реакции Rb шарнирно-неподвижной опоры в точке В, определенное в результате проведения предварительного дополнительного расчета.

При использовании ПК такая дополнительная работа является не только ненужной, но даже вредной, ибо она затрудняет проведение дальнейших исследований. Поэтому перед составлением уравнений равновесия для их решения на ПК:

- введенные проекции Xb и Yb неизвестной реакции Rb просто направим в сторону положительного направления осей координат (рис. 2.9а), тогда знак ответа покажет их истинное направление;
- никакого предварительного нахождения реакций опор делать не будем, ибо упрощение аналитических выкладок для на ПК не имеет никакого значения. Искомые усилия в стержнях, как и реакции опор, получатся из численного или символического решения на ПК системы уравнений равновесия любой степени сложности для всех узлов простой плоской фермы;
- для изучения вариации нагрузки (см. п. 2.3.3) или геометрических факторов (см. п. 2.3.4) варьируемые параметры покажем в самой общей постановке в произвольном текущем положении (направление силы P и положение шарнирно-неподвижной опоры в точке А на рис. 2.9б).

Для исходного положения плоской фермы (рис. 2.9а) составим по два уравнения равновесия для сил, сходящихся соответственно в узлах С, К, Е, L, D, А и В (они приведены также последовательно в пособии [22, с. 9-11]), что в результате будет иметь следующий вид:

Узел С:	$\sum X_i = 0;$	1. $P + S_2 \cdot \cos\alpha = 0;$	
	$\sum Y_i = 0;$	2. $-S_1 - S_2 \cdot \sin\alpha = 0;$	
Узел К:	$\sum X_i = 0;$	3. $-S_2' \cdot \cos\alpha - S_3' = 0;$	
	$\sum Y_i = 0;$	4. $S_2' \cdot \sin\alpha - S_6 = 0;$	
Узел Е:	$\sum X_i = 0;$	5. $S_3 + S_5 \cdot \cos\alpha = 0;$	
	$\sum Y_i = 0;$	6. $S_1' - S_4 - S_5 \cdot \sin\alpha = 0;$	
Узел L:	$\sum X_i = 0;$	7. $-S_5' \cdot \cos\alpha - S_7' = 0;$	(2.39)
	$\sum Y_i = 0;$	8. $S_5' \cdot \sin\alpha + S_6' - S_{10} = 0;$	

$$\begin{array}{ll}
\text{Узел D:} & \sum X_i = 0; & 9. S_7 + S_9 \cdot \cos \alpha = 0; \\
& \sum Y_i = 0; & 10. S_4 - S_8 - S_9 \cdot \sin \alpha = 0; \\
\text{Узел A:} & \sum X_i = 0; & 11. S_8 - Ra = 0. \\
\text{Узел B:} & \sum X_i = 0; & 12. -S_9 \cdot \cos \alpha + Xb = 0; \\
& \sum Y_i = 0; & 13. S_9 \cdot \sin \alpha + S_{10} + Yb = 0;
\end{array}$$

Согласно условию значение $P=11$. На рис. 2.9 и рис. 5 [22, с. 10] уже учтена разность направлений штрихованных и не штрихованных реакций связей для каждого стержня. Поэтому вместо векторных уравнений типа (2.4) используются соответствующие скалярные равенства типа (2.5), учитывающие равенство значений этих величин вплоть до знака, так что им можно присвоить один идентификатор: $S_i = S_i' = X_i$.

Рассмотрим подготовку исходных данных для решения этого примера в САВ REDUCE. Соответствие идентификаторов для формализации системы уравнений (2.39) можно представить в следующей форме:

$$\begin{array}{cccccc}
X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \\
S_1=S_1' & S_2=S_2' & S_3=S_3' & S_4=S_4' & S_5=S_5' & S_6=S_6' \\
\hline
X_7 & X_8 & X_9 & X_{10} & X_{11} & X_{12} & X_{13} \\
S_7=S_7' & S_8=S_8' & S_9=S_9' & S_{10}=S_{10}' & Ra & Xb & Yb
\end{array} \tag{2.40}$$

Для удобства составления программ заменим также используемую греческую букву для обозначения угла α ее идентификатором, записанным в латинской транскрипции: $\alpha = \text{ALFA}$.

Перенесем свободные члены, не содержащие неизвестных, в правые части уравнений (2.39), которые с учетом соответствия идентификаторов (2.40) теперь примут следующий формализованный вид (2.41):

$$\begin{array}{l}
1. X_2 \cdot \cos(\text{ALFA}) = -P \\
2. -X_1 - X_2 \cdot \sin(\text{ALFA}) = 0 \\
3. -X_2 \cdot \cos(\text{ALFA}) - X_3 = 0 \\
4. X_2 \cdot \sin(\text{ALFA}) - X_6 = 0 \\
5. X_3 + X_5 \cdot \cos(\text{ALFA}) = 0 \\
6. X_1 - X_4 - X_5 \cdot \sin(\text{ALFA}) = 0 \\
7. -X_5 \cdot \cos(\text{ALFA}) - X_7 = 0 \\
8. X_5 \cdot \sin(\text{ALFA}) + X_6 - X_{10} = 0
\end{array} \tag{2.41}$$

9. $X_7 + X_9 \cdot \cos(\text{ALFA}) = 0$
10. $X_4 - X_8 - X_9 \cdot \sin(\text{ALFA}) = 0$
11. $X_8 - X_{11} = 0$
12. $-X_9 \cdot \cos(\text{ALFA}) + X_{12} = 0$
13. $X_9 \cdot \sin(\text{ALFA}) + X_{10} + X_{13} = 0$

Для удобства пользования уравнения пронумерованы сверху вниз, начиная с 1, где номером без скобок обозначается порядковый номер строки. Теперь система уравнений (2.41) приобрела явный вид СЛАУ (2.2), в которой выписаны в символьной форме только отличные от нуля элементы. Для дальнейшего её решения или исследования с помощью ПК в системе REDUCE имеются различные возможности.

2.3.2 Решение в численном виде

С увеличением количества уравнений равновесия резко *возрастают достоинства работы с использованием операторов присваивания для ненулевых значений* элементов соответствующих матриц. Они состоят в *сокращении вводимых элементов* (только ненулевых) и в *точном их помещении с указанием места*. Все это значительно облегчает ввод данных и поиск допущенных ошибок.

Мы рассмотрим также для создания вам возможности осознанного выбора и использование операторов MAT. В них элементы матрицы A(I,J) и матрицы-столбца B(I,1) задаются *в полном виде* по строкам.

Сначала представим с использованием многократного оператора присваивания (2.17) значения используемых тригонометрических функций, что является общим для обеих возможностей:

```
SA:=SIN(ALFA):=1/2;                                     22
CA:=COS(ALFA):=SQRT(3)/2;                             24
```

Для задания элементов матрицы A(I,J) и матрицы-столбца B(I,1) в *полном виде* по строкам в операторе MAT их следует дополнить нулями до полных форм $N \times N$ и $N \times 1$, что будет с использованием идентификаторов многократных операторов присваивания 22–24 иметь следующий вид:

```
A:=MAT((0,CA,0,0,0,0,0,0,0,0,0,0,0),                35
(-1,-SA,0,0,0,0,0,0,0,0,0,0,0),
(0,-CA,-1,0,0,0,0,0,0,0,0,0,0),
(0,SA,0,0,0,-1,0,0,0,0,0,0,0),
(0,0,1,0,CA,0,0,0,0,0,0,0,0),
```


щим образом (по вопросам записи программы 2.4 в файле PR2-4 в подкаталоге \C1 корневого каталога текущего диска и запуске ее на выполнение см. пример в п. 1.4.2):

COMMENT **Программа 2.4**: решение типового примера C-1 на REDUCE с использованием многократных операторов присваивания 35-55 для *ненулевых значений* элементов матрицы A и матрицы-столбца B с представлением данных ввода и результатов в численном виде;

```

OUT "\C1\PR2-4.LIS";                                10
ON NERO;                                             15
N:=13;                                              20
SA:=SIN(ALFA):=1/2;                                22
CA:=COS(ALFA):=SQRT(3)/2;                          24
P:=11;                                              29
MATRIX A(N,N), B(N,1), X(N,1);                    30
A(1,2):=CA; A(2,1):=-1; A(2,2):=-SA; A(3,2):=-CA; 35
A(3,3):=-1; A(4,2):=SA; A(4,6):=-1; A(5,3):=1;   37
A(5,5):=CA; A(6,1):=1; A(6,4):=-1; A(6,5):=-SA; 40
A(7,5):=-CA; A(7,7):=-1; A(8,5):=SA; A(8,6):=1; 42
A(8,10):=-1; A(9,7):=1; A(9,9):=CA; A(10,4):=1; 45
A(10,8):=-1; A(10,9):=-SA; A(11,8):=1;          47
A(11,11):=-1; A(12,9):=-CA; A(12,12):=1;        48
A(13,9):=SA; A(13,10):= 1; A(13,13):=1;         50
B(1,1):=-P;                                        55
OFF NERO;                                           80
X:=A**(-1)*B;                                       85
SHUT "\C1\PR2-4.LIS";                               95
;END;                                              99

```

Команда 10 открывает выводной файл PR2-4.LIS, находящийся в подкаталоге \C1 корневого каталога текущего диска, куда будут записываться результаты работы программы 2.4.

Команда 15 запрещает печать нулевых значений при контрольной распечатке матриц A и B, что достигается включением флага NERO.

Оператор присваивания 20 задает значение используемой вспомогательной переменной N, равное количеству тринадцати уравнений равновесия в данной задаче.

Многokратные операторы присваивания 22 и 24 задают значения тригонометрических функций угла α . В них каждое выражение принимает значение, вычисленное для самой правой части.

Используя такую возможность, операторы присваивания 35–55 ввода данных программы 2.4 можно представить в виде, облегчающем их запись. Теперь вместо тригонометрических функций просто используются переменные, идентификаторы которых для простоты состоят из первых букв обозначений самой функции и используемого угла.

Оператор присваивания 29 задает значение силе P из условия задачи, делая ее связанной переменной при записи элементов матрицы-столбца B (первых частей уравнений равновесия (2.41)).

Оператор 30 описывает матричные переменные с явным заданием ее размерностей:

- матрицу A с числом строк N и столбцов $N=13$, для которой обеспечивается резервирование памяти для хранения $13 \times 13 = 169$ значений элементов A ,
- матрицы-столбцы B и X , содержащих по $N=13$ элементов каждый.

Операторы присваивания 35 – 50 и 55 вводят ненулевые элементы соответственно матрицы A и матрицы-столбца B в символьной форме, которые задаются с использованием обозначений для переменных из операторов 22 и 24. Контрольная печать введенных исходных данных будет получена автоматически применением терминатора ";".

Команда 80 разрешает печать нулевых значений перед решением СЛАУ, что достигается выключением флага NERO. Это сделано для возможности распечатки всех значений матрицы-столбца X .

Оператор 85 выполняет непосредственное решение СЛАУ задачи статики и печать выходной информации (значений элементов матрицы-столбца $X(I,1)$).

Команда 95 закрывает выводной файл PR2-4.LIS, находящийся в подкаталоге \C1 корневого каталога текущего диска, куда записывались результаты работы программы 2.4, а команда 99 стандартным способом заканчивает файл PR2-4, предназначенный для считывания.

Результаты решения после окончания работы программы 2.4 (вместе со всей предусмотренной в ней печатью) окажутся в файле PR2-4.LIS, находящемся в подкаталоге \C1 корневого каталога текущего диска, и будут иметь для рассматриваемого примера следующие значения:

$$\begin{aligned}
X(1,1) &:= \frac{11}{\sqrt{3}} & X(2,1) &:= -\frac{22}{\sqrt{3}} \\
X(3,1) &:= 11 & X(4,1) &:= \frac{22}{\sqrt{3}} \\
X(5,1) &:= -\frac{22}{\sqrt{3}} & X(6,1) &:= -\frac{11}{\sqrt{3}} \\
X(7,1) &:= 11 & X(8,1) &:= \frac{33}{\sqrt{3}} \\
X(9,1) &:= -\frac{22}{\sqrt{3}} & X(10,1) &:= -\frac{22}{\sqrt{3}} \\
X(11,1) &:= \frac{33}{\sqrt{3}} & X(12,1) &:= -11 \\
X(13,1) &:= \frac{33}{\sqrt{3}}
\end{aligned} \tag{2.44}$$

Расположение элементов матриц на экране дисплея, как отмечалось в п. 2.1, будет зависеть от используемой версии REDUCE.

Дополнение 2.9. Ввод исходных данных можно также организовать с использованием операторов MAT 35 и 55, в которых элементы матрицы $A(I,J)$ и матрицы-столбца $B(I,1)$ задаются в *полном виде* по строкам.

Это придется сделать, если вам не хочется отягощать свою память кажущейся вам лишней информацией дополнения 2.1, а применение операторов присваивания для ввода *только ненулевые значения* элементов соответствующих матриц с указанием их расположения вы считаете, в отличие от нас, неудобным.

В этом случае вместо операторов присваивания 35 – 50 и 55 достаточно только представить в программе 2.4 операторы 35 и 55 соответственно по фрагментам (2.42) и (2.43).

Это пример как бы обратного применения дополнения 2.1. Надеемся, что вы теперь в состоянии сами осознанно применить к программе 2.4 и все остальные дополнения 2.2 – 2.8 в прямом или обратном виде, все ближе на практике знакомясь с возможностями CAB REDUCE.

Например, для представления результатов решения в приближенном виде с использованием действительных чисел, что важно для сравнения полученных результатов с вашим аналитическим решением, нужно использовать дополнение 2.4. Для этого в программе 2.4 следует:

- установить флаги BIGFLOAT и NUMVAL оператором 87:
ON BIGFLOAT, NUMVAL; 87
- распечатать результаты решения оператором 88:
X; 88
- отменить установленные на время режимы оператором 89:
OFF BIGFLOAT, NUMVAL; 89

После работы такой дополненной программы 2.4 результаты решения (2.44) предстанут в виде действительных чисел (только вместо идентификатора X будет использоваться MAT):

MAT(1,1):= 6.351039261	MAT(2,1):= - 12.70207852
MAT(3,1):= 11	MAT(4,1):= 12.70207852
MAT(5,1):= - 12.70207852	MAT(6,1):= - 6.351039261
MAT(7,1):= 11	MAT(8,1):= 19.05311778 (2.45)
MAT(9,1):= - 12.70207852	MAT(10,1):= - 12.70207852
MAT(11,1):= 19.05311778	MAT(12,1):= -11
MAT(13,1):= 19.05311778	

2.3.3 Исследование влияния вариации нагрузки

Исследуем влияние вариации нагрузки на значения реакций опор и усилий в стержнях плоской шарнирно-стержневой конструкции, изображенной на рис. 2.9а.

Сначала *исследуем их зависимость от значения силы P*. Для этого сделаем ее свободной переменной, удалив из программы 2.4 оператор 29, задающий силе P конкретное значение, а также очистив для надежности ее значение командой

CLEAR P; 29

Теперь после работы измененной программы 2.4 мы получим значения усилий в стержнях рассматриваемой плоской конструкции в зависимости от значения силы P :

$$\begin{aligned}
 X(1,1) &:= \frac{P}{\sqrt{3}} & X(2,1) &:= -\frac{2 * P}{\sqrt{3}} \\
 X(3,1) &:= P & X(4,1) &:= \frac{2 * P}{\sqrt{3}} \\
 X(5,1) &:= -\frac{2 * P}{\sqrt{3}} & X(6,1) &:= -\frac{P}{\sqrt{3}} \\
 X(7,1) &:= P & X(8,1) &:= \frac{3 * P}{\sqrt{3}} & (2.46) \\
 X(9,1) &:= -\frac{2 * P}{\sqrt{3}} & X(10,1) &:= -\frac{2 * P}{\sqrt{3}} \\
 X(11,1) &:= \frac{3 * P}{\sqrt{3}} & X(12,1) &:= -P \\
 X(13,1) &:= \frac{3 * P}{\sqrt{3}}
 \end{aligned}$$

Их графики представляют собой прямые линии, выходящие из начала координат, с тангенсами углов наклона, равным коэффициентам перед силой P в соотношениях (2.46).

Чтобы исследовать влияние вариации угла поворота силы на значения реакций опор и усилий в стержнях рассматриваемой плоской шарнирно-стержневой конструкции, силу P нужно представить в произвольном положении.

Для этого ее следует от горизонтального положения на рис. 2.9а, которое будем считать начальным, повернуть против часовой стрелки на небольшой угол γ (рис. 2.9б).

В результате уравнения равновесия 1 и 2 системы (2.39) примут вид:

1. $P * \cos \gamma + S_2 * \cos \alpha = 0$;
2. $P * \sin \gamma - S_1 - S_2 * \sin \alpha = 0$;

Выполним вышеописанные действия по их формализации:

- заменим используемые греческие буквы для обозначения углов α и γ их идентификаторами, записанными в латинской транскрипции ($\alpha = \text{ALFA}$, $\gamma = \text{GAMMA}$),
- перенесем свободные члены, не содержащие неизвестных, в правые части уравнений 1 и 2, которые с учетом соответствия идентификаторов (2.40) теперь примут следующий измененный вид в системе (2.41):

$$1. X_2 * \cos(\text{ALFA}) = -P * \cos(\text{GAMMA})$$

$$2. -X_1 - X_2 * \sin(\text{ALFA}) = -P * \sin(\text{GAMMA})$$

Теперь оператор 55 программы 2.4 должен быть представлен в следующей форме:

$$B(1,1) := -P * \cos(\text{GAMMA}); \quad B(2,1) := -P * \sin(\text{GAMMA}); \quad 55$$

Имена со скобками $\sin(\text{GAMMA})$ и $\cos(\text{GAMMA})$ можно условно рассматривать в качестве свободных переменных, так как в программе 2.4 им не присваивалось никаких значений.

Для большей надежности работы программы предварительно очистим их командой CLEAR:

$$\text{CLEAR SIN}(\text{GAMMA}), \text{COS}(\text{GAMMA}); \quad 27$$

В результате работы такой измененной программы 2.4 мы получим значения реакций опор и усилий в стержнях плоской конструкции в зависимости от вариации направления постоянной по модулю силы P , определяемого изменением угла γ (рис. 2.9б):

$$X(1,1) := \frac{11 * (\cos(\text{gamma}) + \text{sqrt}(3) * \sin(\text{gamma}))}{\text{sqrt}(3)}$$

$$X(2,1) := - \frac{22 * \cos(\text{gamma})}{\text{sqrt}(3)}$$

$$X(3,1) := 11 * \cos(\text{gamma})$$

$$X(4,1) := \frac{11 * (2 * \cos(\text{gamma}) + \text{sqrt}(3) * \sin(\text{gamma}))}{\text{sqrt}(3)}$$

$$X(5,1) := - \frac{22 * \cos(\text{gamma})}{\text{sqrt}(3)}$$

$$\begin{aligned}
X(6,1) &:= - \frac{11 * \cos(\gamma)}{\sqrt{3}} \\
X(7,1) &:= 11 * \cos(\gamma) \\
X(8,1) &:= \frac{11 * (3 * \cos(\gamma) + \sqrt{3}) * \sin(\gamma)}{\sqrt{3}} \\
X(9,1) &:= - \frac{22 * \cos(\gamma)}{\sqrt{3}} \\
X(10,1) &:= - \frac{22 * \cos(\gamma)}{\sqrt{3}} \\
X(11,1) &:= \frac{11 * (3 * \cos(\gamma) + \sqrt{3}) * \sin(\gamma)}{\sqrt{3}} \\
X(12,1) &:= - 11 * \cos(\gamma) \\
X(13,1) &:= \frac{33 * \cos(\gamma)}{\sqrt{3}}
\end{aligned}
\tag{2.47}$$

Нижеследующие графики наглядно показывают эти зависимости. Как видно из их сравнения, значения реакций опор и усилий в стержнях плоской конструкции, изображенной на рис. 2.9б, при вращении постоянной по модулю силы P в той же плоскости также изменяются по синусоидальному закону:

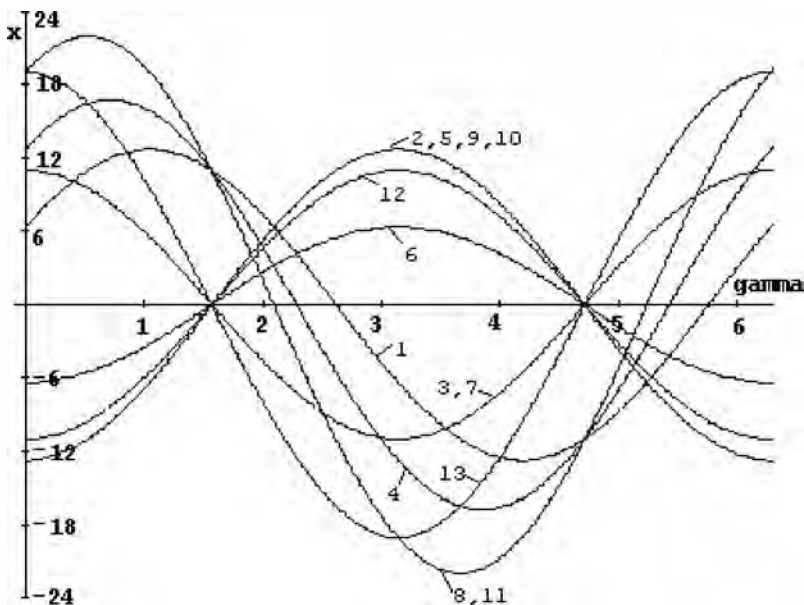


Рис. 2.10. Зависимости усилий в стержнях и реакций опор (X , кН) плоской конструкции (рис. 2.9б) от изменения угла GAMMA (рад) при вращении силы P : 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6 , 7 – S_7 , 8 – S_8 , 9 – S_9 , 10 – S_{10} , 11 – R_a , 12 – X_b , 13 – Y_b

2.3.4 Исследование влияния вариации геометрических факторов

Изучим влияние вариации геометрических факторов рассматриваемой плоской фермы на примере изменения положения ее левой шарнирно-неподвижной опоры в точке A , с которой связан промежуточный элемент в виде 8-го стержня. Его положение при вариации будет определяться углом φ (с идентификатором FI), отсчитываемом от вертикальной оси (начального положения) против часовой стрелки (рис. 2.9б). Точка A при этом вместе с 8-м стержнем будет вращаться вокруг точки D , а угол φ изменяться от 0 до 2π радиан.

Реакция R_a будет также направлена вдоль 8-го стержня при любом его положении. Однако теперь для определения ее модуля и направления

удобнее ввести проекции X_a и Y_a , направленные в сторону положительного направления осей координат (рис. 2.9б).

Теперь для составления уравнений равновесия, позволяющих исследовать влияние вариации положения левой шарнирно-неподвижной опоры, рассматриваемую плоскую ферму нужно представить в произвольном положении. Для этого 8-й стержень следует от вертикального положения на рис. 2.9а повернуть против часовой стрелки на небольшой угол φ (рис. 2.9б). В результате изменятся уравнения равновесия 9-11 системы (2.39) и к ней добавится 14-е уравнение:

$$\begin{aligned}
 9. \quad S_7 + S_9 \cdot \cos \alpha + S_8 \cdot \sin \varphi &= 0; \\
 10. \quad S_4' - S_8 \cdot \cos \varphi - S_9 \cdot \sin \alpha &= 0; \\
 11. \quad X_a - S_8' \cdot \sin \varphi &= 0; \\
 14. \quad Y_a + S_8' \cdot \cos \varphi &= 0;
 \end{aligned} \tag{2.48}$$

Внесем также изменения в соответствие идентификаторов (2.40), пожив в них $X_{11} = X_a$ и $X_{14} = Y_a$. Введем обозначения $\alpha = \text{ALFA}$, $\varphi = \text{FI}$. С учетом этого формализация системы (2.48) будет иметь следующий вид:

$$\begin{aligned}
 9. \quad X_7 + X_9 \cdot \cos(\text{ALFA}) + X_8 \cdot \sin(\text{FI}) &= 0; \\
 10. \quad X_4 - X_8 \cdot \cos(\text{FI}) - X_9 \cdot \sin(\text{ALFA}) &= 0; \\
 11. \quad X_{11} - X_8 \cdot \sin(\text{FI}) &= 0; \\
 14. \quad X_{14} + X_8 \cdot \cos(\text{FI}) &= 0;
 \end{aligned} \tag{2.49}$$

Теперь в программе 2.4 следует соответствующим образом изменить предложения 20 и 45–52, задающие с помощью операторов присваивания значение N и ненулевые элементы фрагмента (2.49), где жирным шрифтом выделены отличающиеся элементы:

```

N:=14;                                20
A(8,10):=-1; A(9,7):=1; A(9,9):=CA;   A(10,4):=1;      45
A(10,8):=-COS(FI); A(10,9):=-SA; A(11,8):=-SIN(FI);  47
A(11,11):=1; A(12,9):=-CA; A(12,12):=1;      48
A(13,9):=SA; A(13,10):= 1; A(13,13):=1;      50
A(9,8):=SIN(FI); A(14,8):=COS(FI); A(14,14):=1;    52

```

Очистим для большей надежности работы программы имена со скобками SIN(FI) и COS(FI) командой CLEAR:

```

CLEAR SIN(FI), COS(FI);                    27

```


В результате работы такой измененной программы 2.4 мы получим значения усилий в стержнях плоской конструкции в зависимости от вариации положения ее левой шарнирно-неподвижной опоры:

$$\begin{aligned}
 X(1,1) &:= \frac{11 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(2,1) &:= - \frac{22 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(3,1) &:= 11 \qquad X(4,1) := \frac{22 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(5,1) &:= - \frac{22 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(6,1) &:= - \frac{11 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \qquad (2.50) \\
 X(7,1) &:= 11 \qquad X(8,1) := \frac{33}{\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi})} \\
 X(9,1) &:= - \frac{22 * (\text{sqrt}(3) * \cos(\text{fi}) + 2 * \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(10,1) &:= - \frac{22 * (\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi}))}{3 * \cos(\text{fi}) - \text{sqrt}(3) * \sin(\text{fi})} \\
 X(11,1) &:= \frac{33 * \sin(\text{fi})}{\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi})} \\
 X(12,1) &:= - \frac{11 * (\text{sqrt}(3) * \cos(\text{fi}) + 2 * \sin(\text{fi}))}{\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi})} \\
 X(13,1) &:= \frac{33 * \cos(\text{fi})}{\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi})} \\
 X(14,1) &:= - \frac{33 * \cos(\text{fi})}{\text{sqrt}(3) * \cos(\text{fi}) - \sin(\text{fi})}
 \end{aligned}$$

Нижеследующие графики наглядно показывают эти зависимости. Как видно из их сравнения, значения усилий в стержнях плоской конструкции при изменении положения опоры в той же плоскости изменяются немонотонным образом. При значениях $\varphi = \pi/3$ и $\varphi = 4\pi/3$ конструкция находится в неустойчивом положении. При этих значениях угла φ происходит потеря равновесия и устойчивости конструкции, что математически находит свое отражение в стремлении к бесконечности соответствующих реакций, в выражениях для которых при $\varphi = \pi/3$ или $\varphi = 4\pi/3$ происходит деление на 0. Это очень хорошо видно на графиках зависимости значений реакций стержней 1–10 и опор 11–14 от величины угла φ , построенных по соотношениям (2.50):

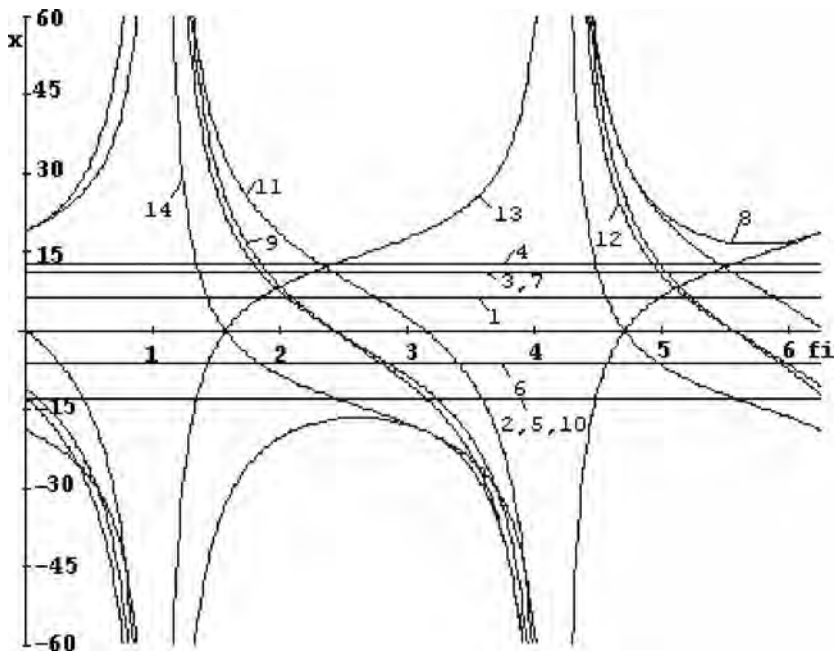


Рис. 2.11. Зависимости усилий в стержнях и реакций опор (X , кН) плоской конструкции (рис. 2.9б) от вариации положения шарнирно-неподвижной опоры в точке A , определяемом углом φ (рад): 1 – S_1 , 2 – S_2 , 3 – S_3 , 4 – S_4 , 5 – S_5 , 6 – S_6 , 7 – S_7 , 8 – S_8 , 9 – S_9 , 10 – S_{10} , 11 – X_a , 12 – X_b , 13 – Y_b , 14 – Y_a

Теперь получим соотношения, описывающие значения реакций при вариации опоры одновременно с изменением положения и модуля действующей силы P .

Для этого следующим образом изменим программу 2.4:

- полностью очистим все используемые переменные и имена со скобками командой 27:

```
CLEAR P, SIN (FI) , COS (FI) , SIN (GAMMA) , COS (GAMMA) ; 27
```

- удалим оператор 29;
- представим оператор 55 в следующей форме:

```
B (1, 1) :=-P * COS (GAMMA) ; B (2, 1) :=-P * SIN (GAMMA) ; 55
```

- изменим соответствующим образом предложения 45–52, задающие с помощью операторов присваивания ненулевые элементы фрагмента (2.49), где жирным шрифтом выделены отличающиеся элементы:

```
A (8, 10) :=-1 ; A (9, 7) :=1 ; A (9, 9) :=CA ; A (10, 4) :=1 ; 45
```

```
A (10, 8) :=-COS (FI) ; A (10, 9) :=-SA ; A (11, 8) :=-SIN (FI) ; 47
```

```
A (11, 11) :=-1 ; A (12, 9) :=-CA ; A (12, 12) :=1 ; 48
```

```
A (13, 9) :=SA ; A (13, 10) := 1 ; A (13, 13) :=1 ; 50
```

```
A (9, 8) :=SIN (FI) ; A (14, 8) :=COS (FI) ; A (14, 14) :=1 ; 52
```

Теперь после работы такой модифицированной программы 2.4 получим соотношения (2.51), описывающие значения реакций при вариации опоры одновременно с изменением положения и модуля действующей силы P .

Как уже отмечалось, выражения такого общего вида позволяют легко получать значения реакций при любом наборе данных для всей конструкции и действующей нагрузки, проверяя наши предположения, сделанные на основании анализа вариаций каких-либо отдельных факторов:

$$X(1,1) = (p * (\sqrt{3}) * \cos(fi) * \cos(\gamma) + 3 * \cos(fi) * \sin(\gamma) - \cos(\gamma) * \sin(fi) - \sqrt{3} * \sin(fi) * \sin(\gamma)) / (3 * \cos(fi) - \sqrt{3} * \sin(fi))$$

$$X(2,1) = - \frac{2 * \cos(\gamma) * p * (\sqrt{3}) * \cos(fi) - \sin(fi)}{3 * \cos(fi) - \sqrt{3} * \sin(fi)}$$

$$\begin{aligned}
X(3,1) &:= \cos(\gamma) * p, \\
X(4,1) &:= (p * (2 * \sqrt{3} * \cos(\beta) * \cos(\gamma) + 3 * \cos(\beta) * \sin(\gamma) - \\
&\quad 2 * \cos(\gamma) * \sin(\beta) - \sqrt{3} * \sin(\beta) * \sin(\gamma))) / \\
&\quad (3 * \cos(\beta) - \sqrt{3} * \sin(\beta)) \\
X(5,1) &:= - \frac{2 * \cos(\gamma) * p * (\sqrt{3} * \cos(\beta) - \sin(\beta))}{3 * \cos(\beta) - \sqrt{3} * \sin(\beta)} \\
X(6,1) &:= - \frac{\cos(\gamma) * p * (\sqrt{3} * \cos(\beta) - \sin(\beta))}{3 * \cos(\beta) - \sqrt{3} * \sin(\beta)} \\
X(7,1) &:= \cos(\gamma) * p \tag{2.51} \\
X(8,1) &:= \frac{p * (3 * \cos(\gamma) + \sqrt{3} * \sin(\gamma))}{\sqrt{3} * \cos(\beta) - \sin(\beta)} \\
X(9,1) &:= - (2 * p * (\sqrt{3} * \cos(\beta) * \cos(\gamma) + 2 * \cos(\gamma) * \sin(\beta) + \\
&\quad \sqrt{3} * \sin(\beta) * \sin(\gamma))) / (3 * \cos(\beta) - \sqrt{3} * \sin(\beta)) \\
X(10,1) &:= - \frac{2 * \cos(\gamma) * p * (\sqrt{3} * \cos(\beta) - \sin(\beta))}{3 * \cos(\beta) - \sqrt{3} * \sin(\beta)} \\
X(11,1) &:= \frac{\sin(\beta) * p * (3 * \cos(\gamma) + \sqrt{3} * \sin(\gamma))}{\sqrt{3} * \cos(\beta) - \sin(\beta)} \\
X(12,1) &:= - (p * (\sqrt{3} * \cos(\beta) * \cos(\gamma) + 2 * \cos(\gamma) * \sin(\beta) + \\
&\quad \sqrt{3} * \sin(\beta) * \sin(\gamma))) / (\sqrt{3} * \cos(\beta) - \sin(\beta)) \\
X(13,1) &:= \frac{p * (3 * \cos(\beta) * \cos(\gamma) + \sin(\beta) * \sin(\gamma))}{\sqrt{3} * \cos(\beta) - \sin(\beta)} \\
X(14,1) &:= - \frac{\cos(\beta) * p * (3 * \cos(\gamma) + \sqrt{3} * \sin(\gamma))}{\sqrt{3} * \cos(\beta) - \sin(\beta)}
\end{aligned}$$

2.4 МНОВОВАРИАНТНЫЕ ЗАДАЧИ

2.4.1 Формализация уравнений

В качестве примера многовариантной задачи статики со “средним” количеством уравнений равновесия воспользуемся аналитическим решением типового задания по определению реакций опор составных конструкций с внутренними односторонними связями для системы трех тел (*задание С-9 [23, с. 50-59]*). Его расчетная схема приведена на рис. 2.12а.

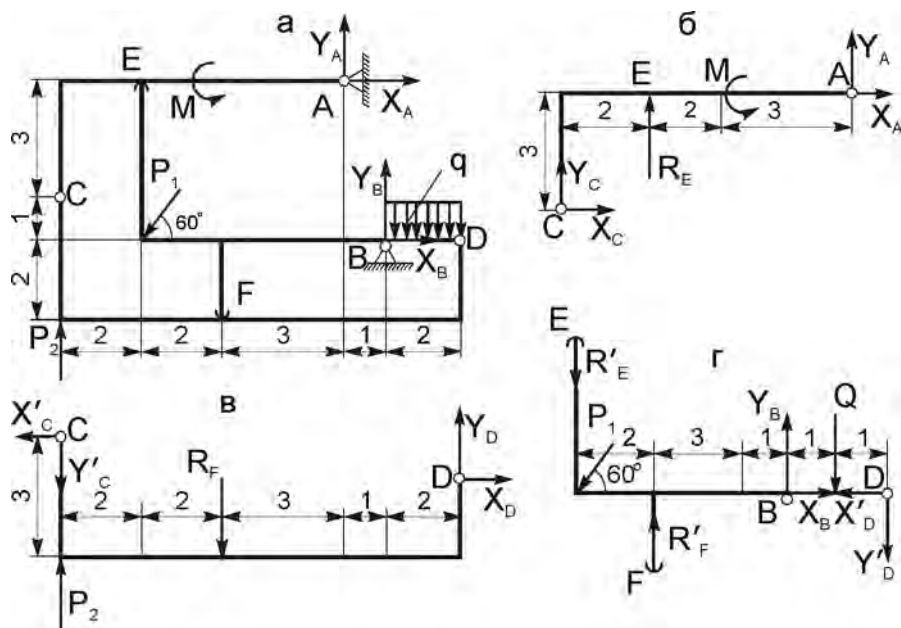


Рис. 2.12. Расчетная схема плоской составной конструкции с внутренними односторонними связями (а) и ее составных частей (б–г)

В задаче рассматривается два случая, когда “работает” какая-либо одна из односторонних связей: R_e или R_f , а вторая равна нулю (рис. 2.12б–г). Соответственно этому необходимо решить на ПК две системы

уравнений равновесия (14) и (24) [23, с. 56 и 57], которые имеют следующий вид:

• **1 случай (Rf = 0):**

1. $Re*2 + M - Xa*3 + Ya*7 = 0,$
2. $Xa + Xc = 0,$
3. $Re + Ya + Yc = 0,$
4. $Xc'*1 + Yc'*10 - P_2*10 = 0,$
5. $-Xc' + Xd = 0,$
6. $-Yc' + P_2 + Yd = 0,$
7. $Re*6 + 6*P_1*\sin60^\circ - Q*1 - Yd'*2 = 0,$
8. $Xb - Xd' - P_1*\cos60^\circ = 0,$
9. $-Re - P_1*\sin60^\circ + Yb - Q - Yd' = 0.$

• **2 случай (Re = 0):**

1. $M - Xa*3 + Ya*7 = 0,$
2. $Xa + Xc = 0,$
3. $Ya + Yc = 0,$
4. $Rf*6 + Xc'*1 + Yc'*10 - P_2*10 = 0,$
5. $-Xc' + Xd = 0,$
6. $-Rf - Yc' + P_2 + Yd = 0,$
7. $-Rf*4 + 6*P_1*\sin60^\circ - Q*1 - Yd'*2 = 0,$
8. $Xb - Xd' - P_1*\cos60^\circ = 0,$
9. $Rf - P_1*\sin60^\circ + Yb - Q - Yd' = 0.$

Здесь уравнения пронумерованы сверху вниз, для каждой СЛАУ начиная с 1, где номером без скобок обозначен порядковый номер строки в системе уравнений.

Рассмотрим подготовку исходных данных для численного решения СЛАУ (2.52) и (2.53), отличающихся друг от друга только использованием одной неизвестной величины: (Re или Rf). Составим соответствие идентификаторов для СЛАУ (2.52), присвоив значение реакции $Re=Re'=X_1$. Для СЛАУ (2.53) оно будет таким же, только X_1 вместо Re приравняется к Rf: $Rf=Rf'=X_1$.

Соответствие идентификаторов для СЛАУ (2.52):

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	
$Re=Re'$	Xa	Ya	Xb	Yb	$Xd=Xd'$	$Yd=Yd'$	$Xc=Xc'$	$Yc=Yc'$	(2.54)

Для удобства составления программ обозначим также идентификатором ALFA используемый угол 60° : $ALFA = 60^\circ$. Перепишем СЛАУ (2.52)–

(2.53) с учетом таблицы идентификаторов, перенеся все члены, не содержащие неизвестных, в правую часть и подсчитаем их с учетом условия задачи ($P_1=10$ кН, $P_2=3$ кН, $M=20$ кН.м, $Q=2$ кН):

1. $2 \cdot X_1 - 3 \cdot X_2 + 7 \cdot X_3 = -M = -20,$
2. $X_2 + X_8 = 0,$
3. $X_1 + X_3 + X_9 = 0,$
4. $X_8 + 10 \cdot X_9 = 10 \cdot P_2 = 30,$
5. $X_6 - X_8 = 0,$
6. $X_7 - X_9 = -P_2 = -3,$
7. $6 \cdot X_1 - 2 \cdot X_7 = Q - 6 \cdot P_1 \cdot \sin(\text{ALFA}) = 2 - 30 \cdot \sqrt{3},$
8. $X_4 - X_6 = P_1 \cdot \cos(\text{ALFA}) = 5,$
9. $-X_1 + X_5 - X_7 = Q + P_1 \cdot \sin(\text{ALFA}) = 2 + 5 \cdot \sqrt{3};$

1. $-3 \cdot X_2 + 7 \cdot X_3 = -20,$
2. $X_2 + X_8 = 0,$
3. $X_3 + X_9 = 0,$
4. $6 \cdot X_1 + X_8 + 10 \cdot X_9 = 30,$
5. $X_6 - X_8 = 0,$
6. $-X_1 + X_7 - X_9 = -3,$
7. $-4 \cdot X_1 - 2 \cdot X_7 = Q - 6 \cdot P_1 \cdot \sin(\text{ALFA}) = 2 - 30 \cdot \sqrt{3},$
8. $X_4 - X_6 = 5,$
9. $X_1 + X_5 - X_7 = Q + P_1 \cdot \sin(\text{ALFA}) = 2 + 5 \cdot \sqrt{3}.$

2.4.2 Решение в виде рациональных и вещественных чисел

Представим формализованные СЛАУ (2.55) и (2.56) в матричной форме соответственно в виде:

$$(A_1) (X_1) = B_1, \quad (2.57)$$

$$(A_2) (X_2) = B_2. \quad (2.58)$$

Теперь системы уравнений (2.55) и (2.56) соответственно приобрели явный вид СЛАУ (2.57) и (2.58), в которой выписаны только отличные от нуля элементы. Из их сравнения видно, что матрицы A_1 и A_2 отличаются только первым столбцом (“работает” реакция Re или Rf), а матрицы-столбцы B_1 и B_2 совпадают, так как нагрузка не меняется.

Для дальнейшего их решения или исследования с помощью ПК в системе REDUCE имеются различные возможности. Рассмотрим сначала их самое простое решение, когда матрицы A_1 и A_2 задаются независимо друг от друга, причем покажем как использование операторов присваива-

ния только для ненулевых значений элементов, так и применение для этой цели операторов MAT.

Для задания только ненулевых значений элементов соответствующих матриц $A_1(I,J)$, $A_2(I,J)$ и матрицы-столбца $B_1(I,1)$ следует организовать ввод исходных данных с использованием операторов присваивания. Предварительно следует определить согласно дополнению 2.1 положение и значения всех ненулевых элементов матриц A_1 , A_2 и B_1 .

Применение операторов присваивания позволяет вводить только их ненулевые значения с указанием их расположения. При этом нулевые значения можно не указывать, так как все элементы матриц $A_1(I,J)$, $A_2(I,J)$ и матрицы-столбца $B_1(I,1)$, размерность которых была описана оператором 30, равны 0. Поэтому достаточно только задать в программе 2.5 их ненулевые элементы. Теперь структура простейшей программы п. 2.1 для решения двухвариантных систем уравнений (2.55) и (2.56) в системе REDUCE может быть реализована, например, следующим образом (по вопросам записи программы 2.5 в файле PR2-5 в подкаталоге \C9 корневого каталога текущего диска и запуске ее на выполнение см. п. 7 и пример в п. 1.4.2):

COMMENT **Программа 2.5:** решение типового примера C-9 на REDUCE с описанием только ненулевых элементов матриц A_1 , A_2 и матрицы-столбца B_1 , копированием $B_2=B_1$ с представлением данных ввода и результатов в численном виде;

```

OUT "\C9\PR2-5.LIS";                                10
ON NERO;                                             15
N:=9;                                               20
SIN(ALFA) := SQRT(3)/2;                             22
COS(ALFA) := 1/2;                                   24
P1:=10; P2:=3; Q:=2;                                29
MATRIX A1(N,N), B1(N,1), X1(N,1),                 30
      A2(N,N), B2(N,1), X2(N,1);
A1(1,1) := 2; A1(1,2) := -3; A1(1,3) := 7; A1(2,2) := 1;    35
A1(2,8) := 1; A1(3,1) := 1; A1(3,3) := 1; A1(3,9) := 1;    37
A1(4,8) := 1; A1(4,9) := 10; A1(5,6) := 1; A1(5,8) := -1;  40
A1(6,7) := 1; A1(6,9) := -1; A1(7,1) := 6; A1(7,7) := -2;  42
A1(8,4) := 1; A1(8,6) := -1; A1(9,1) := -1;            45
A1(9,5) := 1; A1(9,7) := -1;                          47
B1(1,1) := -20; B1(4,1) := 30; B1(6,1) := -P2;        55

```



```

B1(7,1):=Q-6*P1*SIN(ALFA); B1(8,1):= P1*COS(ALFA);      57
B1(9,1):=Q+P1*SIN(ALFA);                                  58
A2(1,2):=-3; A2(1,3):= 7; A2(2,2):= 1; A2(2,8):=1;      60
A2(3,3):= 1; A2(3,9):= 1; A2(4,1):= 6; A2(4,8):=1;      62
A2(4,9):=10; A2(5,6):= 1; A2(5,8):=-1; A2(6,1):=-1;     65
A2(6,7):= 1; A2(6,9):=-1; A2(7,1):=-4; A2(7,7):=-2;    67
A2(8,4):= 1; A2(8,6):=-1; A2(9,1):=1;                   70
A2(9,5):= 1; A2(9,7):=-1;                                 72
B2:=B1;                                                     75
OFF NERO;                                                   80
X1:=A1**(-1)*B1;                                           85
X2:=A2**(-1)*B2;                                           86
SHUT "\C9\PR2-5.LIS";                                       95
END;                                                         99

```

Команда 10 открывает выводной файл PR2-5.LIS, находящийся в подкаталоге \C9 корневого каталога текущего диска, куда будут записываться результаты работы программы 2.5.

Команда 15 запрещает печать нулевых значений при контрольной распечатке матриц A_1 , A_2 и матриц-столбцов B_1 и B_2 , что достигается включением флага NERO.

Оператор присваивания 20 задает значение используемой вспомогательной переменной N , равное количеству девяти уравнений равновесия в данной задаче.

Операторы присваивания 22 и 24 задают значения тригонометрических функций угла α , обозначенного посредством идентификатора ALFA.

Операторы присваивания 29 задают значения силам P_1 , P_2 и Q из условия задачи, делая их связанными переменными при записи элементов матрицы-столбца B_1 (правых частей уравнений равновесия (2.55)).

Оператор 30 описывает матричные переменные с явным заданием их размерностей:

- матрицы A_1 и A_2 с числом строк N и столбцов $N=9$, для которых обеспечивается резервирование памяти для хранения $9 \times 9 = 81$ значений элементов для каждой из них,
- матрицы-столбцы B_1 , B_2 и X_1 , X_2 содержащие по $N=9$ элементов каждый.

Операторы присваивания 35 – 50 и 55 – 58 вводят ненулевые элементы соответственно матрицы A_1 и матрицы-столбца B_1 в символьной форме, которые задаются с использованием обозначений для переменных из операторов 22 и 24. Контрольная печать введенных исходных данных будет получена автоматически применением терминатора ";".

Операторы 60 – 72 аналогично присваивают ненулевым элементам матрицы A_2 соответствующие значения.

Оператор присваивания 75 приравнивает между собой матрицы-столбца B_2 и B_1 , так как нагрузка не меняется и они одинаковы. В результате этого каждый элемент B_2 получает значение B_1 . Такой элегантный и простой способ копирования матриц лишней раз подчеркивает достоинства системы REDUCE.

В любом из языков программирования высокого уровня, например Фортране, такая операция может быть выполнена только с использованием соответствующего цикла, который с помощью арифметического оператора присваивания для всех $N=9$ элементов матриц-столбцов присвоит значение каждого элемента $B_1(I,1)$ соответствующему элементу $B_2(I,1)$: $B_2(1,1) = B_1(1,1)$, $B_2(2,1) = B_1(2,1)$, ..., $B_2(9,1) = B_1(9,1)$.

Команда 80 разрешает печать нулевых значений перед решением СЛАУ, что достигается выключением флага NERO. Это сделано для возможности распечатки всех значений матриц-столбцов X_1 и X_2 .

Операторы 85 и 86 выполняют непосредственное решение СЛАУ (2.55) и (2.56) соответственно и печать выходной информации (значений элементов матриц-столбцов $X_1(I,1)$ и $X_2(I,1)$).

Команда 95 закрывает выводной файл PR2-5.LIS, находящийся в подкаталоге \C9 корневого каталога текущего диска, куда записывались результаты работы программы 2.5, а команда 99 стандартным способом заканчивает файл PR2-5, предназначенный для считывания.

Для задания элементов матриц $A_1(I,J)$, $A_2(I,J)$ и матрицы-столбца $B_1(I,1)$ в полном виде по строкам

- уравнения (2.55) и (2.56) следует дополнить нулями до полных форм 9×9 и 9×1 ;
- из программы 2.5 нужно удалить операторы 35 – 50, 55 – 58 и 60 – 72;

- после этого все значения элементов используемых матриц следует задать по строкам в соответствующих операторах MAT, что будет иметь для программы 2.5 следующий вид:

```

A1:=MAT( (2, -3, 7, 0, 0, 0, 0, 0, 0),           35
(0, 1, 0, 0, 0, 0, 0, 1, 0),
(1, 0, 1, 0, 0, 0, 0, 0, 1),
(0, 0, 0, 0, 0, 0, 0, 1, 10),
(0, 0, 0, 0, 0, 1, 0, -1, 0),
(0, 0, 0, 0, 0, 0, 1, 0, -1),
(6, 0, 0, 0, 0, 0, -2, 0, 0),
(0, 0, 0, 1, 0, -1, 0, 0, 0),
(-1, 0, 0, 0, 1, 0, -1, 0, 0));           (2.59)
B1:=MAT( (-20), (0), (0), (30), (0), (-P2),     55
(Q-6*P1*SIN(ALFA)), (P1*COS(ALFA)), (Q+P1*SIN(ALFA)));
A2:=MAT( (0, -3, 7, 0, 0, 0, 0, 0, 0),           60
(0, 1, 0, 0, 0, 0, 0, 1, 0),
(0, 0, 1, 0, 0, 0, 0, 0, 1),
(6, 0, 0, 0, 0, 0, 0, 1, 10),
(0, 0, 0, 0, 0, 1, 0, -1, 0),
(-1, 0, 0, 0, 0, 0, 1, 0, -1),
(-4, 0, 0, 0, 0, 0, -2, 0, 0),
(0, 0, 0, 1, 0, -1, 0, 0, 0),
(1, 0, 0, 0, 1, 0, -1, 0, 0));

```

В обоих случаях результаты решения после окончания работы программы 2.5 (вместе со всей предусмотренной в ней печатью) окажутся в файле PR2-5.LIS, находящемся в подкаталоге \C9 корневого каталога текущего диска, и будут иметь для рассматриваемого примера следующие значения:

$$\begin{aligned}
 X1(1,1) &:= -\frac{3*(185*\sqrt{3}-12)}{116} & X1(2,1) &:= \frac{5*(75*\sqrt{3}-8)}{58} \\
 X1(3,1) &:= \frac{2*(60*\sqrt{3}-47)}{29} & X1(4,1) &:= -\frac{15*(25*\sqrt{3}-22)}{58} \\
 X1(5,1) &:= \frac{5*(5*\sqrt{3}+13)}{29} & X1(6,1) &:= -\frac{5*(75*\sqrt{3}-8)}{58}
 \end{aligned}$$

$$\begin{aligned}
X1(7,1) &:= \frac{75 * \text{sqrt}(3) - 8}{116} & X1(8,1) &:= - \frac{5 * (75 * \text{sqrt}(3) - 8)}{58} \\
X1(9,1) &:= \frac{5 * (15 * \text{sqrt}(3) + 68)}{116} & & (2.60) \\
X2(1,1) &:= \frac{185 * \text{sqrt}(3) - 12}{31} & X2(2,1) &:= \frac{2 * (105 * \text{sqrt}(3) - 11)}{31} \\
X2(3,1) &:= \frac{2 * (45 * \text{sqrt}(3) - 49)}{31} & X2(4,1) &:= - \frac{3 * (70 * \text{sqrt}(3) - 59)}{31} \\
X2(5,1) &:= \frac{65 * \text{sqrt}(3) + 67}{31} & X2(6,1) &:= - \frac{2 * (105 * \text{sqrt}(3) - 11)}{31} \\
X2(7,1) &:= \frac{95 * \text{sqrt}(3) - 7}{31} & X2(8,1) &:= - \frac{2 * (105 * \text{sqrt}(3) - 11)}{31} \\
X2(9,1) &:= - \frac{2 * (45 * \text{sqrt}(3) - 49)}{31}
\end{aligned}$$

Используем дополнение 2.4 и представим результаты решения (2.60) в приближенном виде с использованием действительных чисел, что нужно для сравнения полученных результатов с вашим аналитическим или численным решением на ПК. Для этого в программу 2.5 следует внести следующие дополнения:

- установить флаги BIGFLOAT и NUMVAL оператором 87:
ON BIGFLOAT, NUMVAL; 87
- распечатать результаты решения операторами 88–89:
X1; 88
X2; 89
- отменить установленные на время режимы оператором 90:
OFF BIGFLOAT, NUMVAL; 90

После работы такой дополненной программы 2.5 результаты решения (2.60) предстанут в виде действительных чисел (только вместо идентификаторов X1 и X2 в обоих случаях будет использоваться MAT), что мы и покажем в нижеследующей распечатке, соответствующей работе операторов 88–89:

X1;

$$\begin{array}{ll} \text{MAT}(1,1):= - 7.976379296 & \text{MAT}(2,1):= 10.50862067 \\ \text{MAT}(3,1):= 3.925517234 & \text{MAT}(4,1):= - 5.50862068 \\ \text{MAT}(5,1):= 3.734482752 & \text{MAT}(6,1):= - 10.50862067 \\ \text{MAT}(7,1):= 1.050862067 & \text{MAT}(8,1):= - 10.50862067 \\ \text{MAT}(9,1):= 4.050862062 & \end{array} \quad (2.61)$$

X2;

$$\begin{array}{ll} \text{MAT}(1,1):= 9.949031743 & \text{MAT}(2,1):= 11.02322523 \\ \text{MAT}(3,1):= 1.867096643 & \text{MAT}(4,1):= - 6.023225674 \\ \text{MAT}(5,1):= 5.792903577 & \text{MAT}(6,1):= - 11.02322567 \\ \text{MAT}(7,1):= 5.081935718 & \text{MAT}(8,1):= - 11.02322567 \\ \text{MAT}(9,1):= - 1.867096717 & \end{array}$$

2.4.3 Оператор цикла FOR ... DO. Учет особенностей задания C-9

В САВ REDUCE оператор цикла **FOR ... DO** позволяет указанное число раз повторить нужную последовательность выражений и команд при различных значениях некоторой переменной, называемой счетчиком цикла. Он может быть записан в следующей общей форме:

$$\text{FOR } I:= N1 \text{ STEP } N2 \text{ UNTIL } N3 \text{ DO оператор} \quad (2.62)$$

где: I — неиндексированная целая переменная, соответствующая последовательно изменяющейся в цикле величине и называемая параметром или счетчиком цикла;

$N1, N2, N3$ — константы, неиндексированные переменные или арифметические выражения, имеющие целочисленные значения, причем $N1$ и $N3$ — соответственно начальное и конечное значения параметра I , а $N2$ — шаг его изменения при каждом прохождении цикла.

Оператор, стоящий после **DO**, выполняется сначала при значении целой переменной $I=N1$, затем при $I=N1+N2$, затем при $I=N1+2*N2$,

$I=N1+3*N2$ и т.д., пока значение I меньше или равно $N3$. Как только станет значение параметра цикла $I=N2+N3$, то выполнение цикла прекращается и программа переходит к выполнению оператора, следующего непосредственно за оператором цикла.

Если конечное значение $N3$ меньше (а для отрицательных значений шага $N2$ больше), чем начальное значение $N1$, то оператор цикла вообще не выполняется. *Оператор FOR ... DO также не предназначен для использования в качестве выражения: его значение в этом случае равно нулю.*

После DO должен стоять один оператор. Если же необходимо использовать несколько операторов, применяют групповой оператор.

Конструкцию $I:= N1 \text{ STEP } N2 \text{ UNTIL } N3$ обычно называют заголовком цикла.

Если $N2=1$, то в этом случае $\text{STEP } 1 \text{ UNTIL}$ можно заменить двусторонним и общая форма оператора цикла FOR ... DO (2.62) примет упрощенный вид:

$\text{FOR } I:= N1:N3 \text{ DO оператор}$ (2.63)

Например, в результате выполнения цикла:

$\text{FOR } I:= 1:N \text{ DO } A2(I, 1):=0;$ (2.64)

будут заданы нулевые значения всем N элементам первого столбца матрицы A_2 . Конечно, значение переменной N , присутствующей в заголовке цикла, обязательно должно быть предварительно задано в программе (например, оператором присваивания $20 (N:=9)$ в программе 2.5).

Замечание. Значение переменной, используемой в качестве параметра цикла, не связано с ее значением вне цикла.

Поэтому в качестве имени параметра цикла можно использовать I , хотя вне цикла I есть мнимая единица.

Таким образом, выполнение оператора $\text{FOR } I:= \dots$ не меняет заложенное в систему REDUCE соотношение $I**2=-1$.

Теперь можно учесть еще одну особенность задания С-9, рекомендованного в сборнике [23] для применения ЭВМ к решению задач статики.

Она заключается в том, что матрицы A_1 и A_2 отличаются только первым столбцом (см. уравнения (2.52)-(2.53)). Это позволяет почти наполо-

вину сократить необходимые вводимые данные. Для этого в базовой программе 2.5 нужно:

- удалить операторы присваивания 60–72;
- предусмотреть после ввода матрицы A_1 ее копирование в рабочий массив A_2 :

```
A2:=A1; 60
```

- затем следует обнулить первый столбец матрицы A_2 , например, с использованием цикла (2.64):

```
FOR I:= 1:N DO A2 (I, 1) :=0; 62
```

- теперь нужно ввести только отличающиеся ненулевые элементы 1-го столбца матрицы A_2 (2.53).

```
A2 (4, 1) :=6; A2 (6, 1) :=-1; A2 (7, 1) :=-4; A2 (9, 1) :=1; 65
```

После работы программы 2.5 с вышеописанными изменениями результаты решения СЛАУ (2.52)-(2.53) предстанут, конечно, в том же виде (2.60)

Дополнение 2.10. Результат выполнения оператора цикла 62 не контролируется пользователем, так как он не распечатывается при выполнении модифицированной программы 2.5. Чтобы получить печать выполнения оператора из цикла, нужно использовать команду WRITE. В ней в качестве параметра должен стоять нужный оператор присваивания:

```
FOR I:= 1:N DO WRITE A2 (I, 1) :=0; 62
```

Однако мы и теперь не получим желаемого результата: нулевые значения первого столбца матрицы A_2 так и не будут распечатаны. После несколько недоуменного изучения программы 2.5 мы вскоре поймем причину: оператор 62 стоит до выключения режима NERO, запрещающего печать нулевых значений.

Поэтому команду 80, которая разрешает печать нулевых значений перед решением СЛАУ, выключая флаг NERO, следует расположить впереди оператора 62:

```
OFF NERO; 61
```

Вот теперь нужная контрольная печать будет получена: все девять нулевых значений первого столбца матрицы A_2 будут распечатаны *по одному в каждой строке*: $A_2(1,1)=0, \dots, A_2(9,1)=0$.

Заметим, что циклы с операцией DO WRITE используются, как правило, для распечатки массивов переменных.

2.4.4 Исследование вариации нагрузки

В рассматриваемом типовом примере сила P_1 представлена на рис. 2.18 в произвольном положении, определяемом углом $\alpha = \text{ALFA} = 60^\circ$. Он отсчитывается против часовой стрелки от горизонтальной прямой, идущей вправо. Поэтому для исследования влияния вариации угла поворота силы P_1 на значения реакций опор рассматриваемой составной конструкции тригонометрические функции используемого угла ALFA следует сделать в программе 2.5 свободными переменными, очистив их командой CLEAR:

```
CLEAR SIN(ALFA), COS(ALFA);
```

27

В результате работы такой измененной программы 2.5 мы получим значения реакций опор рассматриваемой системы тел в зависимости от вариации направления постоянной по модулю силы P_1 для двух вариантов работы конструкции:

- **1 случай ($R_f = 0$):**

$$\begin{aligned} X1(1,1) &:= -\frac{3 * (185 * \sin(\text{alfa}) - 6)}{58} & X1(2,1) &:= \frac{5 * (75 * \sin(\text{alfa}) - 4)}{29} \\ X1(3,1) &:= \frac{2 * (120 * \sin(\text{alfa}) - 47)}{29} \\ X1(4,1) &:= \frac{5 * (58 * \cos(\text{alfa}) - 75 * \sin(\text{alfa}) + 4)}{29} & (2.65) \\ X1(5,1) &:= \frac{5 * (10 * \sin(\text{alfa}) + 13)}{29} & X1(6,1) &:= -\frac{5 * (75 * \sin(\text{alfa}) - 4)}{29} \\ X1(7,1) &:= \frac{75 * \sin(\text{alfa}) - 4}{58} & X1(8,1) &:= -\frac{5 * (75 * \sin(\text{alfa}) - 4)}{29} \\ X1(9,1) &:= \frac{5 * (15 * \sin(\text{alfa}) + 34)}{58} \end{aligned}$$

Нижеследующие графики наглядно показывают эти зависимости. Как видно из их сравнения, значения реакций опор рассматриваемой составной конструкции при вращении постоянной по модулю силы P_1 в той же плоскости также изменяются по синусоидальному закону:

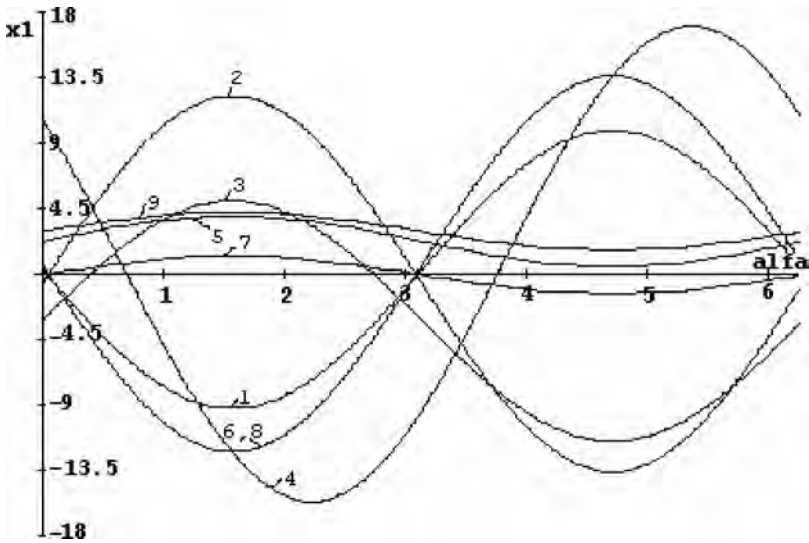


Рис. 2.13. Зависимости реакций опор (X , кН) плоской составной конструкции (рис. 2.18) от изменения угла $ALFA$ (рад) при вращении силы P_1 : 1 – $Re=Re'$, 2 – Xa , 3 – Ya , 4 – Xb , 5 – Yb , 6 – $Xd=Xd'$, 7 – $Yd=Yd'$, 8 – $Xc=Xc'$, 9 – $Yc=Yc'$

• 2 случай ($Re = 0$):

$$X2(1,1) := \frac{2 * (185 * \sin(\text{alfa}) - 6)}{31} \quad X2(2,1) := \frac{2 * (210 * \sin(\text{alfa}) - 11)}{31}$$

$$X2(3,1) := \frac{2 * (90 * \sin(\text{alfa}) - 49)}{31}$$

$$X2(4,1) := \frac{2 * (155 * \cos(\text{alfa}) - 210 * \sin(\text{alfa}) + 11)}{31} \quad (2.66)$$

$$X2(5,1) := \frac{130 * \sin(\text{alfa}) + 67}{31}$$

$$X2(6,1) := - \frac{2 * (210 * \sin(\text{alfa}) - 11)}{31}$$

$$X2(7,1) := \frac{190 * \sin(\text{alfa}) - 7}{31}$$

$$X2(8,1) := - \frac{2 * (210 * \sin(\text{alfa}) - 11)}{31}$$

$$X2(9,1) := - \frac{2 * (90 * \sin(\text{alfa}) - 49)}{31}$$

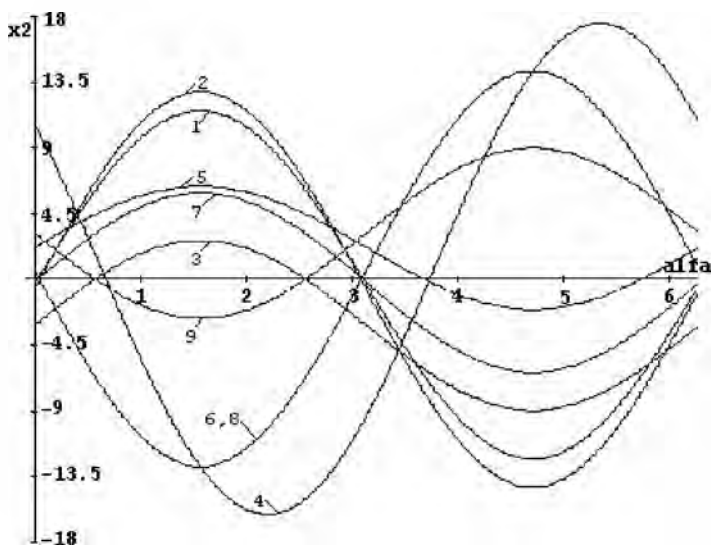


Рис. 2.14. Зависимости реакций опор (X , кН) плоской составной конструкции (рис. 2.18) от изменения угла α (рад) при вращении силы P_1 : 1 – $R_f=R_f'$, 2 – X_a , 3 – Y_a , 4 – X_b , 5 – Y_b , 6 – $X_d=X_d'$, 7 – $Y_d=Y_d'$, 8 – $X_c=X_c'$, 9 – $Y_c=Y_c'$

Как видно из вышеприведенных графиков, синусоидальный характер зависимостей значений реакций опор рассматриваемой составной конструкции при вращении постоянной по модулю силы P_1 не изменяется.

Для удобства дальнейшего анализа выпишем отдельно из (2.65) и (2.66) выражения, определяющие значения реакций $R_e = X_1(1,1)$ и $R_f = X_2(1,1)$, при различных положениях силы P_1 :

$$X_1(1,1) = - \frac{3 * (185 * \sin(\alpha) - 6)}{58}$$

$$X_2(1,1) = \frac{2 * (185 * \sin(\alpha) - 6)}{31}$$

Построим соответствующие графики с использованием системы DE-RIVE [6, 14], передав в нее значения реакций $R_e = X_1(1,1)$ и $R_f = X_2(1,1)$ из (2.65) и (2.66) с помощью буфера обмена (<Ctrl>+<C> — <Ctrl>+<V>):

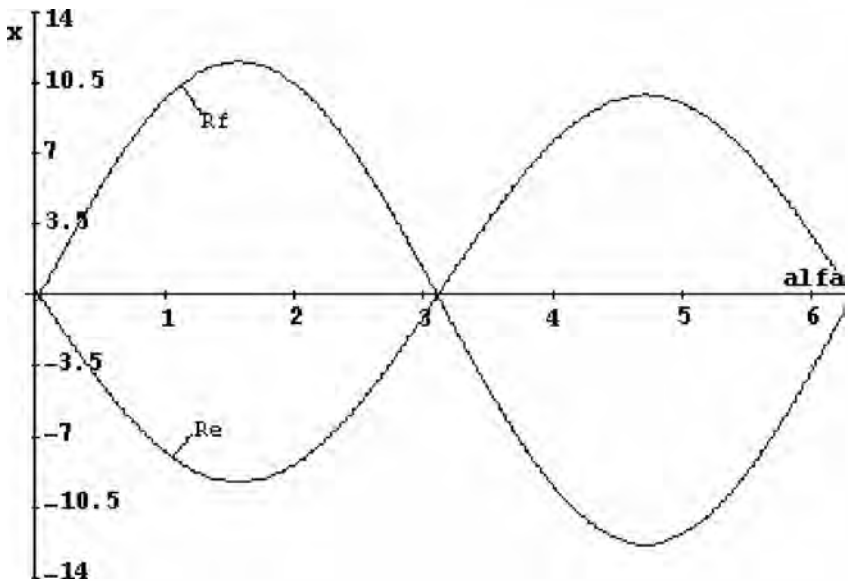


Рис. 2.15. Зависимости реакций опор R_e и R_f (X , кН) плоской составной конструкции с внутренними односторонними связями (рис. 2.18) от изменения угла $ALFA$ (рад) при вращении силы P_1

Из сравнения представленных зависимостей видно, что при значениях угла $ALFA$ от 0 до π “работает” реакция R_f (ее значения положительны, а R_e – отрицательны). При значениях угла $ALFA$ от π до 2π “работает” реакция R_e .

Для исследования влияния вариации угла поворота силы P_2 на значения реакций опор рассматриваемой составной конструкции силу P_2 нужно представить в произвольном положении.

Будем отсчитывать изменяемый при вращении силы P_2 угол β по часовой стрелке от горизонтальной прямой, проходящей через начало силы вправо (рис. 2.18).

Тогда изображенному на рис. 57–58 и 60 [23, с. 55] вертикальному положению силы P_2 будет соответствовать угол $\beta = 90^\circ$, а произвольному положению на рис. 2.18 — угол $\beta < 90^\circ$. После этого уравнения равновесия 4-6 систем (2.52) и (2.53) примут для произвольного положения силы P_2 соответственно следующий вид (2.67) и (2.68):

$$\begin{aligned}
4. & X_c * 1 + Y_c * 10 - P_2 * \cos \beta * 2 - P_2 * \sin \beta * 10 = 0, \\
5. & -X_c + X_d - P_2 * \cos \beta = 0, \\
6. & -Y_c + P_2 * \sin \beta + Y_d = 0.
\end{aligned}
\tag{2.67}$$

и

$$\begin{aligned}
4. & R_f * 6 + X_c * 1 + Y_c * 10 - P_2 * \cos \beta * 2 - P_2 * \sin \beta * 10 = 0, \\
5. & -X_c + X_d - P_2 * \cos \beta = 0, \\
6. & -R_f - Y_c + P_2 * \sin \beta + Y_d = 0.
\end{aligned}
\tag{2.68}$$

Выполним вышеописанные действия по их формализации:

- заменим используемую греческую букву для обозначения угла β ее идентификатором, записанным в латинской транскрипции ($\beta = \text{BETA}$),
- перенесем свободные члены, не содержащие неизвестных, в правые части уравнений 4–6 систем (2.67) и (2.68), которые с учетом соответствия идентификаторов (2.54) теперь примут следующий измененный вид (2.69) и (2.70):

$$\begin{aligned}
4. & X_8 + 10 * X_9 = P_2 * \cos(\text{BETA}) * 2 + P_2 * \sin(\text{BETA}) * 10, \\
5. & X_6 - X_8 = P_2 * \cos(\text{BETA}), \\
6. & X_7 - X_9 = -P_2 * \sin(\text{BETA}),
\end{aligned}
\tag{2.69}$$

и

$$\begin{aligned}
4. & 6 * X_1 + X_8 + 10 * X_9 = P_2 * \cos(\text{BETA}) * 2 + P_2 * \sin(\text{BETA}) * 10, \\
5. & X_6 - X_8 = P_2 * \cos(\text{BETA}), \\
6. & -X_1 + X_7 - X_9 = -P_2 * \sin(\text{BETA}).
\end{aligned}
\tag{2.70}$$

Напомним, что остальные уравнения равновесия 1–3 и 7–9 в системах (2.55) и (2.56) не изменились, а уравнения 4–6 там имеют соответственно вид (2.69) и (2.70).

Теперь оператор 55 программы 2.5 предстанет в измененной форме, а она сама дополнится оператором 56:

```

B1(1,1) := -20; B1(4,1) := P2 * COS(BETA) * 2 + P2 * SIN(BETA) * 10; 55
B1(5,1) := P2 * COS(BETA); B1(6,1) := - P2 * SIN(BETA); 56

```

Теперь наша измененная программа 2.5 учитывает направление силы P_2 в общем виде, определяемом углом β .

Проверим, что для изображенного на рис. 57–58 и 60 [23, с. 55] вертикального положения силы P_2 она дает те же результаты решения (2.60) (или (2.61) при использовании дополнения 2.4).

Для этого дополним программу 2.5 операторами, определяющими значение тригонометрических функций при $\beta = 90^\circ$:

`SIN(BETA) := 1; COS(BETA) := 0;` 26

Убедившись после запуска измененной программы 2.5 в совпадении результатов решения, исследуем влияние вариации направления постоянной по модулю силы P_2 на значения реакций опор рассматриваемой составной конструкции.

Для этого тригонометрические функции используемого угла BETA сделаем в программе 2.5 свободными переменными, очистив их командой CLEAR:

`CLEAR SIN(BETA), COS(BETA);` 27

В результате работы такой измененной программы 2.5 мы получим значения реакций опор рассматриваемой составной конструкции в зависимости от вариации направления постоянной по модулю силы P_2 :

• **1 случай ($R_f = 0$):**

$$X1(1,1) := \frac{3 * (6 * \cos(\beta) - 185 * \sqrt{3}) - 7 * \sin(\beta) + 19}{116}$$

$$X1(2,1) := - \frac{78 * \cos(\beta) - 375 * \sqrt{3} + 315 * \sin(\beta) - 275}{58}$$

$$X1(3,1) := - \frac{2 * (9 * \cos(\beta) - 60 * \sqrt{3}) + 33 * \sin(\beta) + 14}{29}$$

$$X1(4,1) := \frac{3 * (84 * \cos(\beta) - 125 * \sqrt{3}) + 105 * \sin(\beta) + 5}{58}$$

$$X1(5,1) := \frac{18 * \cos(\beta) + 25 * \sqrt{3} - 21 * \sin(\beta) + 86}{29} \quad (2.71)$$

$$X1(6,1) := \frac{252 * \cos(\beta) - 375 * \sqrt{3} + 315 * \sin(\beta) - 275}{58}$$

$$X1(7,1) := \frac{54 * \cos(\beta) + 75 * \sqrt{3} - 63 * \sin(\beta) + 55}{116}$$

$$X1(8,1) := \frac{78 * \cos(\beta) - 375 * \sqrt{3} + 315 * \sin(\beta) - 275}{58}$$

$$X1(9,1) := \frac{54 * \cos(\beta) + 75 * \sqrt{3} + 285 * \sin(\beta) + 55}{116}$$

Нижеследующие графики, представленные на рис. 2.16, показывают эти зависимости.

Как видно из их сравнения, значения реакций опор рассматриваемой составной конструкции при вращении постоянной по модулю силы P_2 в плоскости рисунка также изменяются по синусоидальному закону:

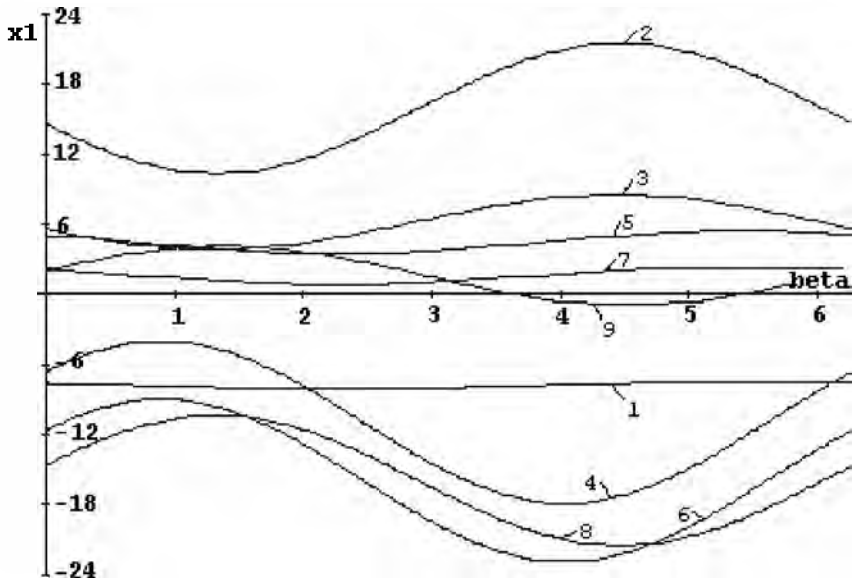


Рис. 2.16. Зависимости реакций опор (X , кН) плоской составной конструкции с внутренними односторонними связями (рис. 2.18) от изменения угла $ALFA$ (рад) при вращении силы P_2 : 1 – $Re=Re'$, 2 – Xa , 3 – Ya , 4 – Xb , 5 – Yb , 6 – $Xd=Xd'$, 7 – $Yd=Yd'$, 8 – $Xc=Xc'$, 9 – $Yc=Yc'$

• **2 случай (Re = 0):**

$$\begin{aligned}
 X2(1,1) &:= - \frac{6 * \cos(\beta) - 185 * \sqrt{3} - 7 * \sin(\beta) + 19}{31} \\
 X2(2,1) &:= - \frac{2 * (21 * \cos(\beta) - 105 * \sqrt{3}) + 84 * \sin(\beta) - 73}{31} \\
 X2(3,1) &:= - \frac{2 * (9 * \cos(\beta) - 45 * \sqrt{3}) + 36 * \sin(\beta) + 13}{31} \\
 X2(4,1) &:= \frac{3 * (45 * \cos(\beta) - 70 * \sqrt{3}) + 56 * \sin(\beta) + 3}{31} \\
 X2(5,1) &:= \frac{18 * \cos(\beta) + 65 * \sqrt{3} - 21 * \sin(\beta) + 88}{31} \quad (2.72) \\
 X2(6,1) &:= \frac{135 * \cos(\beta) - 210 * \sqrt{3} + 168 * \sin(\beta) - 146}{31} \\
 X2(7,1) &:= \frac{12 * \cos(\beta) + 95 * \sqrt{3} - 14 * \sin(\beta) + 7}{31} \\
 X2(8,1) &:= \frac{2 * (21 * \cos(\beta) - 105 * \sqrt{3}) + 84 * \sin(\beta) - 73}{31} \\
 X2(9,1) &:= \frac{2 * (9 * \cos(\beta) - 45 * \sqrt{3}) + 36 * \sin(\beta) + 13}{31}
 \end{aligned}$$

Напомним, что эти зависимости значений реакций опор рассматриваемой составной конструкции в зависимости от вариации направления постоянной по модулю силы P_2 мы получили в результате работы измененной программы 2.5.

Для этого тригонометрические функции используемого угла β в программе 2.5 делались свободными переменными и очищались от своих значений командой CLEAR:

CLEAR SIN(β), COS(β); 27

Зависимости (2.72) реакций опор плоской составной конструкции с односторонними внутренними связями (рис. 2.18) от изменения угла α при вращении силы P_2 представлены на рис. 2.17.

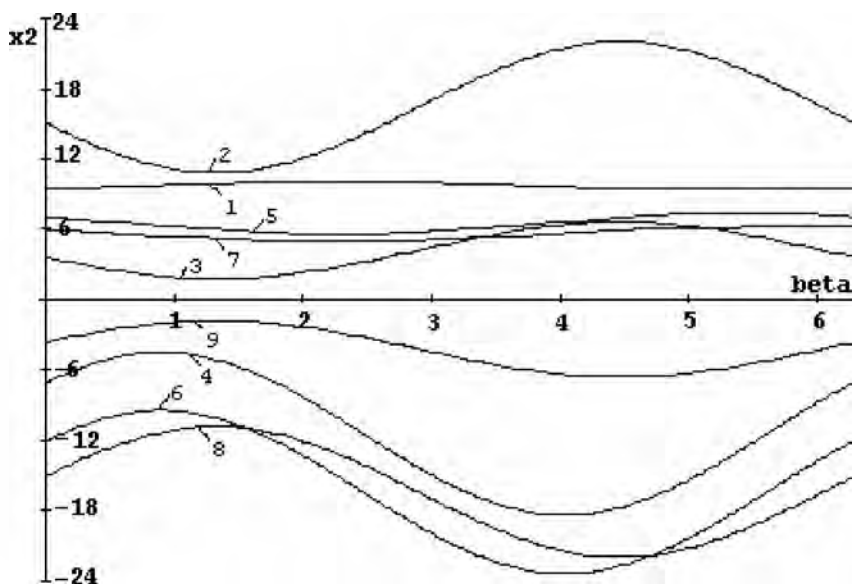


Рис. 2.17. Зависимости реакций опор (X , кН) плоской составной конструкции с внутренними односторонними связями (рис. 2.18) от изменения угла β (рад) при вращении силы P_2 : 1 – $R_f = R_f'$, 2 – X_a , 3 – Y_a , 4 – X_b , 5 – Y_b , 6 – $X_d = X_d'$, 7 – $Y_d = Y_d'$, 8 – $X_c = X_c'$, 9 – $Y_c = Y_c'$

Графики для R_e (рис. 2.16) и R_f (рис. 2.17) весьма характерны: они представляют синусоидальные кривые очень небольшой амплитуды, расположенные в положительной (R_f) или отрицательной (R_e) областях. Поэтому их можно не рассматривать отдельно и сразу сделать вывод, что вариация направления силы P_2 не может изменить знаки односторонних связей: при любых значениях угла β “работает” реакция R_f (ее значения положительны, а R_e – отрицательны).

2.4.5 Исследование вариации геометрических факторов

Для вариации геометрических факторов размеры конструкции, изображенной на рис. 57 [23, с. 55], следует представить в общем виде. Для указанных там величин 1, 2 и 3м положим: $a=1m$, $b=2$, $c=3m$ (рис. 2.18).

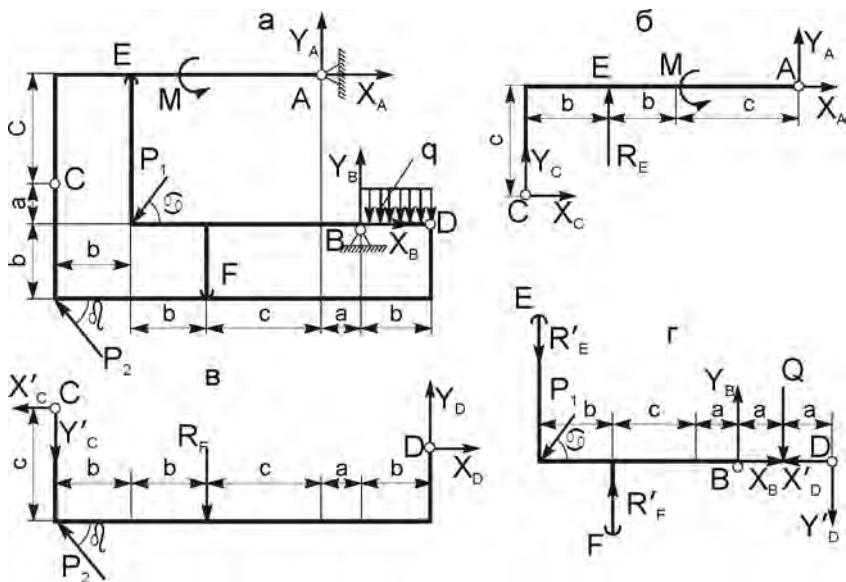


Рис. 2.18. Расчетная схема составной конструкции с внутренними односторонними связями (а) и ее составных частей (б–г), представленная для возможности исследования вариации нагрузки и геометрических факторов

Для удобства составления программ обозначим сразу идентификатором ALFA используемый угол 60° : $ALFA = 60^\circ$.

С учетом используемых обозначений размеров a , b и c перепишем уравнения (2.52)–(2.53) в общем виде:

• **1 случай ($R_f = 0$):**

1. $Re \cdot b + M - X_a \cdot c + Y_a \cdot (2b + c) = 0,$
2. $X_a + X_c = 0,$
3. $Re + Y_a + Y_c = 0,$
4. $X_c \cdot a + Y_c \cdot (3b + a + c) - P_2 \cdot (3b + a + c) = 0,$
5. $-X_c + X_d = 0,$
6. $-Y_c + P_2 + Y_d = 0,$
7. $Re \cdot (a+b+c) + P_1 \cdot \sin(ALFA) \cdot (a+b+c) - q \cdot b \cdot b / 2 - Y_d \cdot b = 0,$
8. $X_b - X_d - P_1 \cdot \cos(ALFA) = 0,$
9. $-Re - P_1 \cdot \sin(ALFA) + Y_b - q \cdot b - Y_d = 0,$

• **2 случай (Re = 0):**

1. $M - X_a * c + Y_a * (2b + c) = 0,$
2. $X_a + X_c = 0,$
3. $Y_a + Y_c = 0,$
4. $R_f * (a+b+c) + X_c * a + Y_c * (3b+a+c) - P_2 * (3b+a+c) = 0,$
5. $-X_c + X_d = 0,$ (2.74)
6. $-R_f - Y_c + P_2 + Y_d = 0,$
7. $-R_f * (a+c) + P_1 * \sin(ALFA) * (a+b+c) - q * b * b / 2 - Y_d * b = 0,$
8. $X_b - X_d - P_1 * \cos(ALFA) = 0,$
9. $R_f - P_1 * \sin(ALFA) + Y_b - q * b - Y_d = 0.$

В уравнениях 7 и 9 систем (2.73)–(2.74) значение сосредоточенной силы Q представлено в виде, явно зависящем от соответствующего размера конструкции b: $Q = q * b.$

С использованием соответствия идентификаторов (2.54) системы уравнений (2.73)–(2.74) примут следующий формализованный вид:

• **1 случай (Rf = 0):**

1. $X_1 * b - X_2 * c + X_3 * (2b + c) = - M,$
2. $X_2 + X_8 = 0,$
3. $X_1 + X_3 + X_9 = 0,$
4. $X_8 * a + X_9 * (3b + a + c) = P_2 * (3b + a + c),$
5. $X_6 - X_8 = 0,$ (2.75)
6. $X_7 - X_9 = - P_2,$
7. $X_1 * (a+b+c) - X_7 * b = q * b * b / 2 - P_1 * \sin(ALFA) * (a+b+c),$
8. $X_4 - X_6 = P_1 * \cos(ALFA),$
9. $-X_1 + X_5 - X_7 = P_1 * \sin(ALFA) + q * b.$

• **2 случай (Re = 0):**

1. $-X_2 * c + X_3 * (2b + c) = - M,$
2. $X_2 + X_8 = 0,$
3. $X_3 + X_9 = 0,$
4. $X_1 * (a+b+c) + X_8 * a + X_9 * (3b+a+c) = P_2 * (3b+a+c),$
5. $X_6 - X_8 = 0,$ (2.76)
6. $-X_1 + X_7 - X_9 = - P_2,$

$$7. - X_1*(a+c) - X_7*b = q*b*b/2 - P_1*\sin\alpha*(a+b+c),$$

$$8. X_4 - X_6 = P_1*\cos\alpha,$$

$$9. X_1 + X_5 - X_7 = q*b + P_1*\sin\alpha.$$

Для вариации геометрических факторов размеров исследуемой конструкции, изображенной на рис. 57 [23, с. 55], выполним нижеследующие действия.

- Определим согласно дополнению 2.1 положение и значение всех ненулевых элементов матриц A_1 и B_1 :

$$A1(1,1):=B; A1(1,2):=-C; A1(1,3):=2*B+C; A1(2,2):=1; \quad 35$$

$$A1(2,8):=1; A1(3,1):=1; A1(3,3):=1; A1(3,9):=1; \quad 37$$

$$A1(4,8):=A; A1(4,9):=3*B+A+C; A1(5,6):=1; A1(5,8):=-1; \quad 40$$

$$A1(6,7):=1; A1(6,9):=-1; A1(7,1):=A+B+C; \quad 42$$

$$A1(7,7):=-B; A1(8,4):=1; A1(8,6):=-1; \quad 45$$

$$A1(9,1):=-1; A1(9,5):=1; A1(9,7):=-1; \quad 47$$

$$B1(1,1):=-M; B1(4,1):=P2*(3*B+A+C); B1(6,1):=-P2; \quad 55$$

$$B1(7,1):=-P1*\sin(ALFA)*(A+B+C)+q*B*B/2; \quad 57$$

$$B1(8,1):=P1*\cos(ALFA); B1(9,1):=P1*\sin(ALFA)+q*B; \quad 58$$

- Учтем особенность выполнения двухвариантной задачи рассматриваемого типового примера C-9 [23, с. 51, 55–59] согласно п. 2.4.3:

- удалим из программы 2.5 операторы присваивания 60–72;

- выполним после ввода матрицы A_1 ее копирование в рабочий массив A_2 :

$$A2:=A1; \quad 60$$

- обнулим первый столбец матрицы A_2 , например, с использованием цикла (2.64):

$$\text{FOR } I:= 1:N \text{ DO } A2(I,1):=0; \quad 62$$

- введем отличающиеся ненулевые элементы 1-го столбца матрицы A_2 (2.76).

$$A2(4,1):=A+B+C; A2(6,1):=-1; A2(7,1):=-C-A; A2(9,1):=1; \quad 65$$

- Зададим геометрические размеры конструкции предложением 18:

$$A:=1; B:=2; C:=3; \quad 18$$

Осознанно скомпоновав приведенные фрагменты, мы получим из программы 2.5 базовую программу 2.6, приспособленную для вариации геометрических размеров конструкции:

```

COMMENT Программа 2.6: решение типового примера С-9 на
REDUCE с описанием только ненулевых элементов матриц A1 и
B1, копированием B2:=B1 и A2:=A1, обнулением 1-го столбца
A2 и заданием ее 4-х отличающихся элементов;
OUT "\C9\PR2-6.LIS"; 10
ON NERO; 15
A:=1; B:=2; C:=3; 18
N:=9; 20
SIN(ALFA) := SQRT(3)/2; 22
COS(ALFA) := 1/2; 24
P1:=10; P2:=3; 29
MATRIX A1(N,N), B1(N,1), X1(N,1), 30
  A2(N,N), B2(N,1), X2(N,1);
A1(1,1):=B; A1(1,2):=-C; A1(1,3):=2*B+C; A1(2,2):=1; 35
A1(2,8):=1; A1(3,1):=1; A1(3,3):=1; A1(3,9):=1; 37
A1(4,8):=A; A1(4,9):=3*B+A+C; A1(5,6):=1; A1(5,8):=-1; 40
A1(6,7):=1; A1(6,9):=-1; A1(7,1):=A+B+C; 42
A1(7,7):=-B; A1(8,4):=1; A1(8,6):=-1; 45
A1(9,1):=-1; A1(9,5):=1; A1(9,7):=-1; 47
B1(1,1):=-M; B1(4,1):=P2*(3*B+A+C); B1(6,1):=-P2; 55
B1(7,1):=-P1*SIN(ALFA)*(A+B+C)+q*B*B/2; 57
B1(8,1):=P1*COS(ALFA); B1(9,1):=P1*SIN(ALFA)+q*B; 58
A2:=A1; 60
FOR I:= 1:N DO A2(I,1):=0; 62
A2(4,1):=A+B+C; A2(6,1):=-1; A2(7,1):=-C-A; A2(9,1):=1; 65
B2:=B1; 75
OFF NERO; 80
X1:=A1**(-1)*B1; 85
X2:=A2**(-1)*B2; 86
SHUT "\C9\PR2-6.LIS"; 95
END; 99

```

Все операторы, команды и предложения программы 2.6 были предва- рительно описаны.

После ее работы результаты решения СЛАУ (2.73)-(2.74) будут запи- саны в файле PR2-6.LIS, находящемся в подкаталоге \C9 корневого ката- лога текущего диска. Их значения, конечно, должны совпасть с результа- тами (2.60).

Убедившись в этом и проверив таким образом работу базовой про- граммы, можно приступать к вариации геометрических размеров конст- рукции **a**, **b** и **c**.

Для изучения отдельного влияния вариации геометрических факто- ров на значения реакций опор нужно из предложения 18 исключить соот- ветствующий оператор присваивания, задающий данный размер конст- рукции **a**, **b** или **c**.

Предварительно нужно очистить исключаемую переменную и сделать свободной, указав ее в качестве параметра команды CLEAR.

При желании выполнить исследование с минимальными переделками программы, можно просто командой CLEAR каждый раз только очищать варьируемую переменную, делая ее свободной, что и мы также будем делать.

Тогда соответствующий идентификатор A, B или C становится сво- бодной переменной и входит в символьном виде в результаты решения, выражая их зависимость от данной величины.

Приведем для рассматриваемого типового примера плоской двухва- риантной системы трех тел [23, с. 51, 55-59] формы команды CLEAR, которой нужно дополнить программу 2.6.

Представим также выборку результатов ее работы в виде соответст- вующих аналитических зависимостей, выражающих влияние отдельного изменения размеров конструкции **a**, **b** и **c** на значения реакций $R_e(X1(1,1))$ и $R_f(X2(1,1))$:

- вариация размера **a**:

$$\begin{aligned} & \text{CLEAR } A; & & 27 \\ X1(1,1) & := - (50*\text{sqrt}(3)*a**2 + 385*\text{sqrt}(3)*a + 675*\text{sqrt}(3) - 18*a - 54) \\ & / (10*a**2 + 87*a + 135)\$ & & (2.77) \end{aligned}$$

$$\begin{aligned} X2(1,1) & := (50*\text{sqrt}(3)*a**2 + 385*\text{sqrt}(3)*a + 675*\text{sqrt}(3) - 18*a - 54) \\ & / (10*a**2 + 71*a + 105)\$ & & (2.78) \end{aligned}$$

- вариация размера **b**:

CLEAR B;

27

$$X1(1,1) := - (110*\sqrt{3}*b^{**2} + 590*\sqrt{3}*b + 600*\sqrt{3}) - 11*b^{**3} - 3*b^{**2} - 22*b) / (4*(6*b^{**2} + 31*b + 30))\$ \quad (2.79)$$

$$X2(1,1) := (110*\sqrt{3}*b^{**2} + 590*\sqrt{3}*b + 600*\sqrt{3}) - 11*b^{**3} - 3*b^{**2} - 22*b) / (2*(8*b^{**2} + 47*b + 60))\$ \quad (2.80)$$

- вариация размера **c**:

CLEAR C;

27

$$X1(1,1) := - (5*\sqrt{3}*c^{**3} + 55*\sqrt{3}*c^{**2} + 140*\sqrt{3}*c + 60*\sqrt{3}) - 2*c^{**2} - 10*c - 24) / (c^{**3} + 11*c^{**2} + 30*c + 16)\$ \quad (2.81)$$

$$X2(1,1) := (5*\sqrt{3}*c^{**3} + 55*\sqrt{3}*c^{**2} + 140*\sqrt{3}*c + 60*\sqrt{3}) - 2*c^{**2} - 10*c - 24) / (c^{**3} + 9*c^{**2} + 22*c + 12)\$ \quad (2.82)$$

Покажем также геометрическую интерпретацию полученных соотношений (2.77) – (2.82):

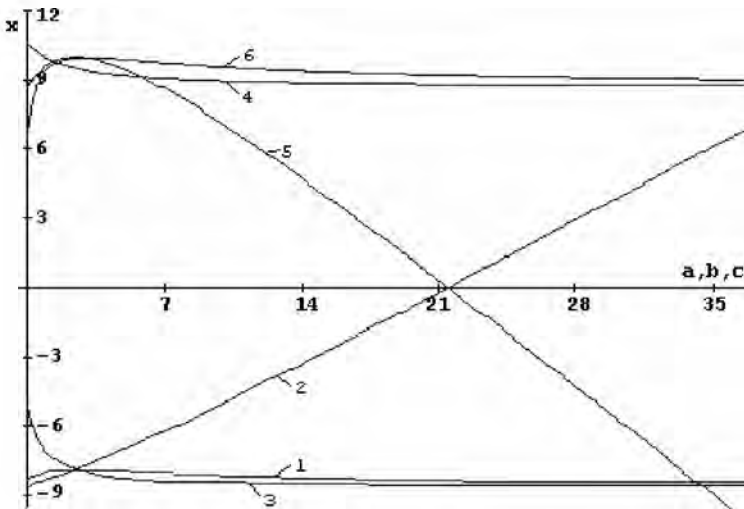


Рис. 2.19. Зависимости величин реакций R_e (кривые 1–3) и R_f (кривые 4–6) от изменения геометрических размеров конструкции **a**, **b** и **c**, изображенной на рис. 57 [23, с. 55], при вариации **a** (кривые 1 и 4), **b** (кривые 2 и 5) и **c** (кривые 3 и 6). Значения R_e и R_f даны в кН, геометрических размеров **a**, **b** и **c** – в метрах

Все графики, как видно из приведенного рисунка, кроме небольшого начального участка носят монотонный характер. Изменение в широких пределах значений **a** и **c** не приводит к изменению знаков реакций R_e и R_f и практически не влияет на их значения (кривые 1, 3–4 и 6).

Изменение же размера **b** приводит к почти постоянному увеличению значений реакции R_e и уменьшению R_f . Это приводит в итоге к изменению условий работы односторонних связей рассматриваемой плоской двухвариантной системы трех тел: вместо R_f начинает “работать” R_e . Однако это происходит при значениях $b > 21$ метра, что редко будет иметь место при практическом использовании.

В этой связи следует заметить, что в настоящем пособии в учебных целях приведены методы решения задач при вариации самых различных факторов: величины и направления действия сил, геометрических размеров и относительного расположения элементов конструкции. *Реальные требования конкретной задачи из инженерной практики определяют необходимость варьирования тех или иных параметров.* Мы пока только изучаем методику проведения аналитического исследования на ПК и программные возможности для этого.

Поэтому в заключение не откажем себе в удовольствии и получим выражения для реакций опор при одновременном изменении всех геометрических размеров конструкции. Для этого очистим в программе 2.6 и сделаем свободными все геометрические размеры **a**, **b** и **c**, указав их в качестве параметров команды CLEAR:

CLEAR A, B, C;

27

Представим выборку результатов работы программы 2.6 в виде соответствующих аналитических зависимостей, выражающих влияние одновременного изменения размеров конструкции **a**, **b** и **c** только на значения реакций R_e ($X1(1,1)$) и R_f ($X2(1,1)$):

$$\begin{aligned}
 X1(1,1) := & -(20*\sqrt{3})*a^2*b + 20*\sqrt{3})*a^2*c + 20*\sqrt{3})*a*b^2 + \\
 & 70*\sqrt{3})*a*b*c + 30*\sqrt{3})*a*c^2 + 30*\sqrt{3})*b^2*c + 40*\sqrt{3})*b*c^2 + \\
 & 10*\sqrt{3})*c^3 - 2*a*b^3 - 2*a*b^2*c + 12*a*b^2 + 6*a*b*c - 40*a*b - 3*b^3*c \\
 & - b^2*c^2) / (2*(2*a^2*b + 2*a^2*c + 3*a*b^2 + 8*a*b*c + 3*a*c^2 + 3*b^2*c + \\
 & 4*b*c^2 + c^3))
 \end{aligned}
 \tag{2.83}$$

$$\begin{aligned}
X1(2,1):= & (10*\sqrt{3}*a^2*b + 10*\sqrt{3}*a^2*c + 40*\sqrt{3}*a*b^2 + \\
& 60*\sqrt{3}*a*b*c + 20*\sqrt{3}*a*c^2 + 30*\sqrt{3}*b^3 + 70*\sqrt{3}*b^2*c + \\
& 50*\sqrt{3}*b*c^2 + 10*\sqrt{3}*c^3 - 12*a^2*b - 6*a^2*c + 40*a^2 - a*b^3 - \\
& a*b^2*c - 48*a*b^2 - 48*a*b*c + 160*a*b - 12*a*c^2 + 80*a*c - 3*b^4 - \\
& 4*b^3*c - 36*b^3 - b^2*c^2 - 66*b^2*c + 120*b^2 - 36*b*c^2 + 160*b*c - 6*c^3 + \\
& 40*c^2) / (2*(2*a^2*b + 2*a^2*c + 3*a*b^2 + 8*a*b*c + 3*a*c^2 + 3*b^2*c + \\
& 4*b*c^2 + c^3))
\end{aligned}
\tag{2.84}$$

Как уже отмечалось, выражения такого общего вида позволяют получать значения реакций при любом наборе данных для всей конструкции, проверяя наши предположения, сделанные на основании анализа вариаций каких-либо отдельных факторов.

3. СИМВОЛЬНОЕ ДИФФЕРЕНЦИРОВАНИЕ В САВ REDUCE И ЕГО ИСПОЛЬЗОВАНИЕ В ЗАДАЧАХ КИНЕМАТИКИ

Решение задач кинематики с применением ПК ранее сводилось к численному дифференцированию уравнений движения, функций положения и т.п. Поэтому рассмотрим различные аспекты применения символьного дифференцирования на ПК в САВ REDUCE.

3.1 ОПЕРАТОР ДИФФЕРЕНЦИРОВАНИЯ DF

Оператор DF используется для выполнения дифференцирования в частных производных по отношению к одной или нескольким переменным. Его синтаксис можно представить в следующей общей форме:

$DF(\text{функция}, \text{аргумент}1, \text{порядок}1, \dots, \text{аргумент}N, \text{порядок}N)$ (3.1)

В (3.1) первым параметром в скобках (функция) является скалярное выражение, которое надо дифференцировать. Остальные параметры определяют независимые переменные (аргумент I), по которым производится дифференцирование, и порядок каждой частной производной (порядок I). В качестве аргумента могут использоваться простые имена или операторы, но не алгебраические выражения в виде их произведений или сумм.

Таким образом, лишь первый параметр оператора дифференцирования может иметь значением алгебраическое выражение. Значениями последующих параметров должны быть либо простого вида переменные, либо целые неотрицательные числа, задающие соответствующий порядок производной, которую требуется взять по стоящей перед цифрой переменной. Если число, выражающее порядок любой частной производной (порядок I) равно единице, то оно может быть опущено. Например:

- для обозначения первой производной нижеследующие два выражения эквивалентны: $df(y, x, 1)$ и $df(y, x)$ означают $\frac{dy}{dx}$;
- для второй и более высоких производных порядок следует указывать обязательно:

- $df(y, x, 2) = \frac{d^2 y}{dx^2}$ и $df(y, x, 2, z, 3, t) = \frac{d^6 y}{dx^2 dz^3 dt}$.

Выполнение оператора DF(Y,X) происходит следующим образом: сначала определяются значения Y и X. Предположим, что аргумент X является свободной переменной, так что его значением является сам символ X. Каждое слагаемое (или другая часть) функции Y, которое зависит от X, будут продифференцированы по обычным правилам. Если Y (или его часть) является выражением, не зависящим от X, то производная Y по X (или ее часть) принимается равной 0.

Если требуется указать на зависимость переменных, то она вводится посредством команды DEPEND, после которой пишутся через запятую параметры. Тем самым устанавливается зависимость первого параметра от всех последующих. Так, запись

depend y, t; df(y5,t);** (3.2)

указывает на то, что Y является функцией T. Поэтому результат дифференцирования будет иметь следующий вид

$$5 * df(y,t) * y^4$$

Зависимость Y от T в явном виде установлена не была, поэтому в результате используется символ DF(Y,T). Введенные зависимости могут быть отменены командой NODEPEND. После

nodepend y, t; df(y5,t);** (3.3)

переменная Y не будет далее зависеть от T, поэтому производная Y по T принимается равной 0.

Рассмотрим применение оператора дифференцирования DF на примерах задач кинематики. Произведем с использованием CAB REDUCE определение кинематических характеристик (скорости, ускорения) и радиуса кривизны траектории материальной точки по заданным уравнениям ее движения. В качестве типовых выберем примеры решения заданий на соответствующие темы: “Кинематика точки” (К-1 [23, с. 60-62] или [22, с. 76-79], К-2 [22, с. 82-87]) и “Сложное движение точки” (К-7 [23, с. 99-106] или К-10 [22, с. 137-143]). Ссылки на задания будем производить по их номеру, причем рассматриваемым примерам присвоим 31-й вариант.

3.2 ПОСТАНОВКА ЗАДАЧ И СТРУКТУРА ПРОГРАММ ПО КИНЕМАТИКЕ ТОЧКИ

Задачи на определение кинематических характеристик являются прекрасной иллюстрацией возможностей символьного дифференцирования системы REDUCE. После составления уравнений движения материальной точки и определения зависимостей ее координат X и Y (а для пространственных задач и Z) от времени T , их нужно дважды продифференцировать по времени для определения проекций скорости и ускорения на соответствующие координатные оси. Еще одно дифференцирование выражения для модуля скорости может быть использовано для определения тангенциального ускорения.

Соответственно этому *структура простейшей программы* для определения радиуса кривизны траектории материальной точки с использованием символьного дифференцирования в системе REDUCE состоит из следующих основных блоков.

1. Открытие командой OUT выводного файла с идентификатором ИДФ (1.26), куда будут записываться результаты работы программы (OUT "идф");. Включение или выключение соответствующих переключателей, устанавливающих нужные режимы печати и представления результатов:
 - выключение режима NAT при получении достаточно длинных результатов для представления их по строкам (OFF NAT;) или включение его при желании представить их в “естественной” форме (ON NAT;). При опущенном флаге NAT все степени печатаются через “**”, для отделения числителя от знаменателя используется символ “/” и в конце каждого предложения ставится терминатор “\$”;
 - объявление вывода в фортран-синтаксисе для удобства проведения дальнейших сложных численных расчетов, что достигается включением флага FORT (ON FORT;). Этот режим включает в себя все достоинства выключения флага NAT при получении длинных результатов для представления их по строкам. Кроме этого строго соблюдается синтаксис фортран-программ:
 - выражения будут начинаться с 7-й колонки;

- если запись не помещается на одной строке, то в последующих строках появится признак продолжения – знак "." (точка) в 6-й колонке (максимальное число строк продолжения 19).
2. Задание значений с помощью операторов присваивания используемым вспомогательным переменным и тригонометрическим функциям. Описание требуемых уравнений движения материальной точки, определяющих зависимость ее координат X и Y (а для пространственных задач и Z) от времени.
 3. Определение проекций скорости точки путем дифференцирования выражений для соответствующих координат по времени. Напомним, что аргументом, по которому производится дифференцирование, должна быть независимая переменная. Переменная T , которой обычно обозначают время, в системе REDUCE является зарезервированной: имеет значение "истина" и используется в символьном режиме. Поэтому применять для обозначения аргумента переменную T возможно, так как он в операторе дифференцирования DF является формальным параметром. Однако *переменной T нельзя присвоить никакого значения*. Это неудобно, ибо делает невозможной проверку получающихся сложных аналитических выражений путем сравнения их численных значений при заданном времени $T1$ с решением без ПК. Поэтому будем:
 - *применять для обозначения аргумента переменную $T1$* ;
 - *очищать ее командой $CLEAR$ перед первым использованием в программе в операторе дифференцирования DF и после присваивания ей численных значений, делая переменную $T1$ свободной*.
 4. Нахождение модуля скорости точки путем извлечения квадратного корня из сумм квадратов проекций скорости.
 5. Определение проекций ускорения точки на координатные оси одним из двух способов:
 - путем взятия *второй производной* от выражений для соответствующих координат по времени $T1$;
 - получением *первой производной* от выражений для соответствующих проекций скорости точки.
 6. Нахождение модуля ускорения точки путем извлечения квадратного корня из сумм квадратов проекций ускорения точки.

7. Определение тангенциального ускорения точки одним из двух способов:
 - получением *первой производной* от соответствующего выражения для скорости точки;
 - использованием обычной формулы для определения тангенциального ускорения по известным значениям проекций скорости и ускорения точки.
8. Определение нормального ускорения точки и радиуса кривизны траектории.
9. Закрытие командой OUT выводного файла с идентификатором ИДФ (1.26), куда записывались результаты работы программы (SHUT "ИДФ");). В конце следует ввести оператор END, что является стандартным способом заканчивать файлы, предназначенные для считывания (END; или для большей надежности END;).

3.3 ОПРЕДЕЛЕНИЕ СКОРОСТИ И УСКОРЕНИЯ ТОЧКИ ПО ЗАДАННЫМ УРАВНЕНИЯМ ЕЕ ДВИЖЕНИЯ

Требуемые для начала решения уравнения движения материальной точки задаются в К-1 ([23, с. 60-62] или [22, с. 76-79]), что позволяет акцентировать внимание на различных аспектах его численного и символьного решения в САВ REDUCE.

Поэтому мы начнем с него показ возможностей символьного дифференцирования.

Уравнения движения материальной точки для типового примера задания К-1 имеют вид ([23, с. 60] или [22, с. 76]):

$$x = 4*t; y = 16*t^2 - 1. \quad (3.4)$$

Для составления простейшей программы по определению кинематических характеристик в системе REDUCE следует реализовать блочную структуру, описанную в п. 3.2.

Это может быть сделано, например, следующим образом (по вопросам записи программы 3.1 в файле PR3-1 в подкаталоге \KIN корневого каталога текущего диска и запуске ее на выполнение см. пример в п. 1.4.2):

```

COMMENT Программа 3.1: Определение скорости, ускорения точки и радиуса кривизны траектории типового примера К-1 ([23, с. 60] или [22, с. 76]) на REDUCE с представлением результатов в символьном виде;
OUT "\KIN\PR3-1.LIS";                                10
CLEAR T1;                                             20
X:=4*T1;                                              25
Y:=16*T1*T1-1;                                       27
VX:=DF(X, T1);                                       30
VY:=DF(Y, T1);                                       32
V:=SQRT(VX*VX+VY*VY);                                40
AX:=DF(VX, T1);                                       50
AY:=DF(VY, T1);                                       52
A:=SQRT(AX*AX+AY*AY);                                60
AT:=(VX*AX+VY*AY)/V;                                  70
AN:=SQRT(A*A-AT*AT);                                  80
RO:=V*V/AN;                                           85
SHUT "\KIN\PR3-1.LIS";                                90
END;                                                  99

```

Номера операторов справа проставлены только для удобства дальнейших пояснений. Этот прием будет использоваться и в дальнейшем.

Напомним, что в программе 3.1 и последующих ее модификациях для удобства первоначальной ориентировки 1-я цифра номера указывает на соответствующий блок п. 3.3: например, операторы 50 и 52 относятся к блоку 5 (определение проекций ускорения точки на координатные оси).

Комментарий, помещенный между зарезервированным словом COMMENT и разделителем в виде знака “точка с запятой”, описывает назначение программы 3.1 и при обработке игнорируется.

Команда 10 открывает выводной файл PR3-1.LIS, находящийся в подкаталоге \KIN корневого каталога текущего диска, куда будут записываться результаты работы программы 3.1. Напомним, что имя файла и подкаталога для ваших программ может быть любым разрешенным идентификатором и должно отражать индивидуальность вашу (для файла) или группы (для подкаталога).

Команда CLEAR 20 для большей надежности очищает переменную T1 перед первым использованием в программе, делая ее свободной, так что ее значением является сам символ T1.

Операторы 25 и 27 присваивают алгебраические выражения, описывающие уравнения движения материальной точки в виде зависимости ее проекций от времени $T1$, переменным X и Y .

Операторы 30 и 32 определяют проекции скорости точки путем дифференцирования алгебраических выражений для соответствующих координат X и Y по времени $T1$. Здесь порядок производной равен единице, поэтому он опущен и не указывается в качестве третьего параметра.

Оператор 40 находит значение модуля скорости точки V путем извлечения квадратного корня из сумм квадратов проекций скорости VX и VY .

Операторы 50 и 52 определяют проекции ускорения точки путем взятия первой производной от алгебраических выражений для соответствующих проекций скорости точки VX и VY по времени $T1$. Здесь порядок производной, равный единице, также не указывается.

Операторы 50 и 52 можно представить также в форме второй производной от соответствующих координат X и Y , где порядок следует указывать обязательно. Это может иметь, например, следующий вид:

$$AX := DF(X, T1, 2); \quad 50$$

$$AY := DF(Y, T1, 2); \quad 52$$

Оператор 60 находит значение ускорения точки A путем извлечения квадратного корня из сумм квадратов проекций ускорений AX и AY .

Оператор 70 определяет тангенциальное ускорение точки AT с использованием обычной формулы по известным значениям проекций скорости и ускорения точки.

Оператор 80 находит значение нормального ускорения точки AN путем извлечения квадратного корня из разности квадратов полного A и тангенциального AT ускорений.

Оператор 85 определяет радиус кривизны траектории RO из формулы для нормального ускорения, деля квадрат скорости точки $V*V$ на ее нормальное ускорение AN .

Команда 90 закрывает выводной файл $PR3-1.LIS$, находящийся в подкаталоге $\backslash KIN$ корневого каталога текущего диска, куда записывались результаты работы программы 3.1.

Оператор 99 запускает специальный учет контроля за файлами, улучшающий эффективность работы системы, что является стандартным способом заканчивать файлы, предназначенные для считывания.

Допустим, что программа 3.1 находится в файле PR3-1 подкаталога \KIN корневого каталога текущего диска. Тогда для запуска ее на выполнение нужно:

- выйти из подкаталога \ KIN и перейти в подкаталог CAB REDUCE с расположенными в нем системными файлами;
- затем следует загрузить систему, для чего нужно запустить на выполнение файл reduce.exe или reduce.bat., совместив с их именами подсветку курсора и нажав клавишу <Enter> (<Ввод>);
- после появления приглашения в командной строке следует ввести:

$$1: \text{in } "\text{KIN}\text{PR3-1}"\$ \quad (3.5)$$

По этой команде в систему будет загружен файл \KIN\PR3-1, выполнены все находящиеся в нем операторы и команды программы 3.1, а результаты их выполнения будут записаны на диске в файле PR3-1.LIS в том же подкаталоге \KIN корневого каталога текущего диска. Они будут иметь для рассматриваемого примера (3.4) следующие значения:

$$\begin{aligned} x &:= 4 * t1 & y &:= 16 * t1^2 - 1 \\ vx &:= 4 & vy &:= 32 * t1 \\ v &:= 4 * \text{sqrt}(64 * t1^2 + 1) & ax &:= 0 \\ ay &:= 32 & a &:= 32 \\ at &:= \frac{256 * t1}{\text{sqrt}(64 * t1^2 + 1)} & an &:= \frac{32}{\text{sqrt}(64 * t1^2 + 1)} \\ ro &:= \frac{\text{sqrt}(64 * t1^2 + 1) * (64 * t1^2 + 1)}{2} \end{aligned} \quad (3.6)$$

Обратим ваше внимание, что команда (3.5) ограничена символом \$, поэтому содержимое выполняемого файла (то есть сама программа PR3-1) в выходном файле PR3-1.LIS не отображается. Это сделано для экономии места. Также с этой же целью результаты представлены по двое в одной строке, а не в каждой по отдельности, как это реально имеет место в выходном файле PR3-1.LIS.

Чтобы получить отображение самой программы PR3-1 в выходном файле, достаточно в команде (3.5) использовать ограничитель ";":

1: in "KIN\PR3-1"; (3.7)

В этом случае после каждого оператора программы PR3-1 будет указан соответствующий результат ее выполнения (3.6), в чем вы легко можете убедиться сами.

Отметим, что даже при использовании команды IN в форме (3.7), предотвратить указанное отображение можно использованием во входном файле (в программе PR3-1) команды

OFF ECHO;

15

В этом случае полученные результаты в выходном файле PR3-1.LIS будут также иметь форму (3.6). Построенные по ним графики зависимостей найденных кинематических характеристик от времени $T1$ будут иметь следующий вид.

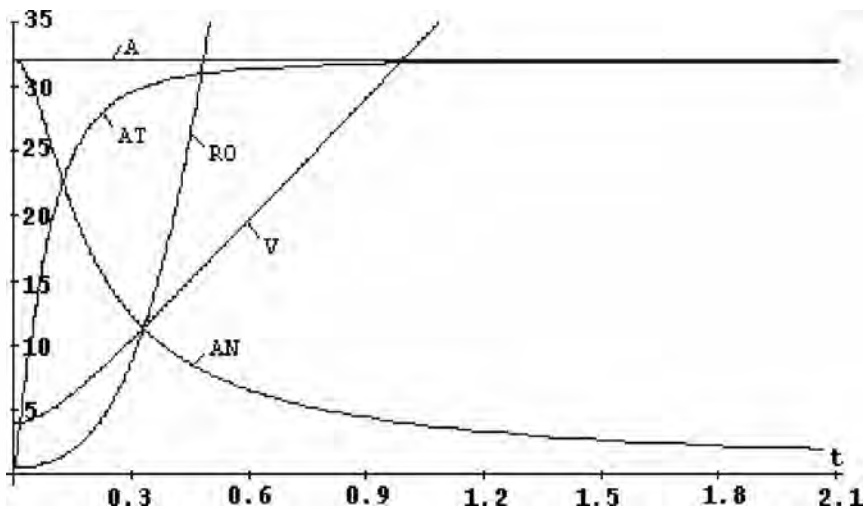


Рис. 3.1. Зависимости кинематических характеристик материальной точки от времени t (с): скорости V (м/с), тангенциального AT , нормального AN и полного ускорения A (м/с²), а также радиуса кривизны траектории RO (м) для типового задания $K-1$ [23, с. 60–62] или [22, с. 76–79].

Дополнение 3.1. Теперь следует дополнить полученное символьное решение (3.6) соответствующим численным при конкретном значении времени T1, заданным в условии задачи. Затем следует убедиться в совпадении результатов работы численного решения дополненной программы 3.1 и вашего аналитического (безмашинного) решения при одинаковом значении времени T1.

Представим результаты решения в численной форме с использованием действительной арифметики согласно дополнения 2.4. Для получения численных значений V, A и RO программу 3.1 следует дополнить ниже-следующим фрагментом.

```
T1:=1/2;                                41
ON BIGFLOAT, NUMVAL;                    42
V;                                       43
OFF BIGFLOAT, NUMVAL; CLEAR T1;        44
T1:=1/2;                                61
ON BIGFLOAT, NUMVAL;                    62
A;                                       63
OFF BIGFLOAT, NUMVAL; CLEAR T1;        64
T1:=1/2;                                86
ON BIGFLOAT, NUMVAL;                    87
RO;                                       88
OFF BIGFLOAT, NUMVAL; CLEAR T1;        89
```

Оператор 41 присваивает заданное в условии типового примера задачи K-1 ([23, с. 60] или [22, с. 76]) значение переменной T1.

Команда 42 устанавливает флаги BIGFLOAT и NUMVAL. Флаг BIGFLOAT обеспечивает использование в многочленах вещественных коэффициентов повышенной точности. Флаг NUMVAL устанавливает режим вычисления значений элементарных функций в форме с плавающей запятой с текущей степенью точности.

Выражение 43 распечатывает результаты численного решения для скорости точки V.

Команда 44 отменяет установленные на время флаги BIGFLOAT и NUMVAL и очищает значение переменной T1, делая ее свободной и пригодной для продолжения символьного решения задачи.

Вышеописанная последовательность операторов 41-44 встречается еще во фрагменте дополнения 3.1 дважды под номерами 61-64 и 86-89,

распечатывая результаты численного решения соответственно для ускорения точки A и радиуса кривизны RO.

Оператор 41 под номерами 61 и 86 присваивает заданное в условии типового примера задания K-1 ([23, с. 60] или [22, с. 76]) значение переменной T1.

Команда 42 под номерами 62 и 87 устанавливает флаги BIGFLOAT и NUMVAL.

Выражения 63 и 88 распечатывают результаты численного решения для ускорения точки A и радиуса кривизны траектории RO соответственно.

Команда 44 под номерами 64 и 89 отменяет установленные на время флаги BIGFLOAT и NUMVAL и очищает значение переменной T1, делая ее свободной и пригодной для продолжения символьного решения задачи.

Результаты численного решения для скорости точки V, ускорения точки A и радиуса кривизны траектории RO для рассматриваемого типового примера будут иметь соответственно следующие значения: 16.4924225, 32 и 35.04639782.

Сравнив их с результатами вашего аналитического (безмашинного) расчета при одинаковом значении времени T1 убеждаемся в правильности решения.

Напомним, что вышеописанный результат достигается только при совместном действии флагов BIGFLOAT и NUMVAL, причем вместо флага BIGFLOAT *не может использоваться* флаг FLOAT.

3.4 СОСТАВЛЕНИЕ УРАВНЕНИЙ ДВИЖЕНИЯ ТОЧКИ И ОПРЕДЕЛЕНИЕ ЕЕ СКОРОСТИ И УСКОРЕНИЯ

Составление уравнения движения материальной точки является необходимым условием для аналитического решения задания K-2 [22, с. 81-87], также как и для его символьного решения на ПК.

Поэтому рассмотрим их получение для типового примера этого задания ([22, с. 82, 86-87]), схема которого представлена на рис. 3.2.

Исходные данные к представленной на рис. 3.2 схеме типового примера задания К-2 имеют вид [22, с. 82]: $\varphi = \pi * t$ (рад), $r = 20$ см, $R = 100$ см, $t_1 = 1/3$ с.

Из условия равенства дуг $OP = MP$ также имеем: $R * \alpha = r * \varphi$ и

$$\alpha = r * \varphi / R = 0.2 * \varphi = 0.2 * \pi * t. \quad (3.8)$$

Чтобы составить уравнения движения материальной точки, нужно для произвольного момента времени T выразить ее координаты через данные задачи и параметр T .

Время T обычно входит в исходные данные (выражения углов или в уравнения движения других точек рассматриваемого механизма).

Полученные таким образом выражения координат для одного произвольного положения материальной точки включают переменную T и остаются справедливыми для любого момента времени, являясь уравнениями движения материальной точки в координатном виде в параметрической форме.

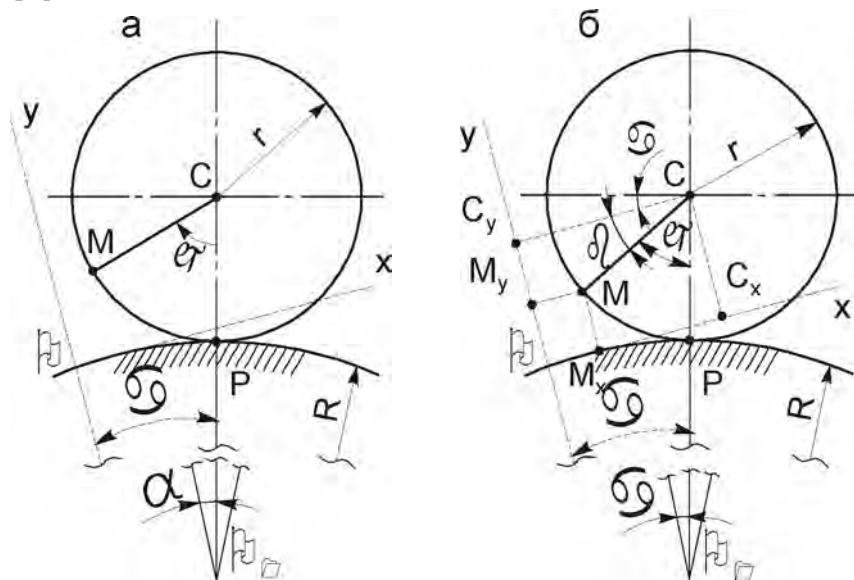


Рис. 3.2. Схема механизма типового примера задания К-2 [22, с. 86]: а — исходное положение; б — дополнения для составления уравнений движения материальной точки M

Условимся, что проекции точек на координатные оси будем обозначать той же буквой с индексом данной оси: M_X, M_Y, C_X, C_Y (см. рис. 3.2б).

Тогда для произвольного положения точки M ее координаты X и Y через данные задачи и параметр T можно выразить следующим образом:

$$\begin{aligned} X &= OM_X = OC_X - C_X M_X = O_1 C * \sin \alpha - CM * \cos \beta = \\ &= (R+r) * \sin \alpha - r * \cos(90^\circ - (\varphi + \alpha)) = (R+r) * \sin \alpha - \\ &- r * \sin(\varphi + \alpha), \end{aligned} \quad (3.9)$$

$$\begin{aligned} Y &= OM_Y = O_1 C_Y - O_1 O - C_Y M_Y = O_1 C * \cos \alpha - R - CM * \sin \beta = \\ &= (R+r) * \cos \alpha - R - r * \sin(90^\circ - (\varphi + \alpha)) = (R+r) * \cos \alpha - R - \\ &- r * \cos(\varphi + \alpha). \end{aligned}$$

Из рис. 3.2б видно, что угол $\beta = 90^\circ - (\varphi + \alpha)$. Поэтому при выводе уравнений движения (3.9) использовано, что

$$\cos \beta = \cos(90^\circ - (\varphi + \alpha)) = \sin(\varphi + \alpha),$$

$$\sin \beta = \sin(90^\circ - (\varphi + \alpha)) = \cos(\varphi + \alpha).$$

Для удобства составления программ заменим также используемые греческие буквы в обозначениях углов их идентификаторами, записанными в их латинской транскрипции:

AL	FI	PI	
α	φ	π	(3.10)

С учетом соотношений (3.10), значений радиусов ($r = 20$ см, $R = 100$ см), углов φ и α (3.8) уравнения движения (3.9) примут следующий вид

$$\begin{aligned} X &= 20 * (6 * \sin(AL) - \sin(FI + AL)), \\ Y &= 20 * (6 * \cos(AL) - \cos(FI + AL) - 5), \end{aligned} \quad (3.11)$$

где $PI = 3.14159265$, $FI = PI * T$, $AL = 0.2 * FI$.

Теперь для решения типового примера задания К-2 в символьном виде в программе 3.1 нужно исправить уравнения движения по фрагменту (3.11). Модифицированную программу 3.1 желательно поместить в другом файле, например, \KIN\PR3-1M. Чтобы не затирались предыдущие результаты расчета, следует также использовать другой файл для записи выходной информации. Для этого в программу-3.1 нужно внести следующие изменения и дополнения, отразив их в комментарии:

COMMENT **Программа 3.1М**: Составление уравнений движения точки и определение ее скорости и ускорения для типового примера К-2 ([22, с. 81-87] на REDUCE с представлением результатов в символьном виде;

```

OUT "\KIN\PR3-1M.LIS";                                10
FI:=PI*T1;                                             23
AL:=0.2*FI;                                           24
X:=20*(6*SIN(AL)-SIN(FI+AL));                         25
Y:=20*(6*COS(AL)-COS(FI+AL)-5);                      27
SHUT "\KIN\PR3-1M.LIS";                               90

```

Команда 10 открывает, а команда 90 закрывает выводной файл PR3-1.LIS, находящийся в подкаталоге \KIN корневого каталога текущего диска, куда записываются результаты работы программы 3.1.

Операторы 23 и 24 присваивают алгебраические выражения, определяющие значения используемых углов FI и AL.

Операторы 25 и 27 присваивают алгебраические выражения, описывающие уравнения движения материальной точки в виде зависимости ее проекций от времени T1, переменным X и Y.

После исправления программы 3.1 с учетом вышеприведенного фрагмента, нужно:

- выйти из рабочего подкаталога \KIN в корневой каталог;
- перейти в подкаталог CAB REDUCE с расположенными в нем системными файлами и загрузить систему;
- после появления приглашения в командной строке следует ввести команду IN в форме (3.5) или (3.7) с указанием имени запускаемого на выполнение файла, например:

1: in "\KIN\PR3-1M"\$ (3.12)

По этой команде в систему будет загружен файл \KIN\PR3-1M, выполнены все находящиеся в нем операторы и команды программы 3.1М, а результаты их выполнения будут записаны на диске в файле PR3-1M.LIS в том же подкаталоге \KIN корневого каталога текущего диска.

Они будут иметь для рассматриваемого примера (3.11) следующие значения.

В целях экономии места приведем только получающиеся аналитические зависимости для:

$$\sin\left(\frac{6 * \pi * t1}{5}\right) + 6 * \sin\left(\frac{6 * \pi * t1}{5}\right)^2 \quad (3.15)$$

Теперь дополним полученное символьное решение (3.13)–(3.15) соответствующим численным при заданном времени T1. Для этого в программе 3.1M, дополненной по фрагменту дополнения 3.1, в операторах 41, 61 и 86 укажем значение времени, заданное в условии задачи T1=1/3 с.

После выполнения такой дополненной программы 3.1M по команде (3.12), результаты численного решения для скорости точки V, ускорения точки A и радиуса кривизны траектории RO для рассматриваемого типового примера будут иметь соответственно следующие значения:

$$V=75.39822369, A=263.767832 \text{ и } RO=34.28571429. \quad (3.16)$$

Сравнив их с результатами вашего аналитического расчета при одинаковом значении времени T1, убеждаемся в правильности решения. После чего с легким сердцем строим по соотношениям (3.13)–(3.15) на ПК с использованием системы DERIVE графики (см. [6, 14]), представленные на рис. 3.3.

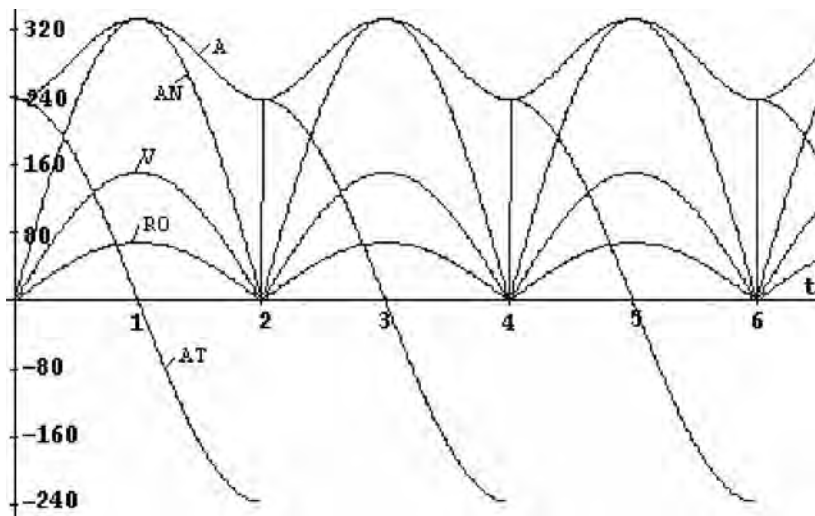


Рис. 3.3. Зависимости кинематических характеристик материальной точки от времени t (с): скорости V (м/с), тангенциального AT , нормального AN и полного ускорения A (м/с²), а также радиуса кривизны траектории RO (м) для типового задания К-2 [22, с. 82, 86–87].

Качественный анализ приведенных кривых подтверждает правильность их построения, ибо выполняются все необходимые соотношения:

- при максимальном значении скорости V тангенциальное ускорение $AT = dV/dt = 0$;
- при любом значении времени t выполняется, что квадрат полного ускорения A равняется сумме квадратов тангенциального и нормального ускорений, что подтверждает визуальная проверка;
- при $t_1 = 1/3$ из графиков рис. 3.3 примерно получаются определенные ранее численные значения (3.16) для V , A и RO ;
- уравнения движения точки (3.11) представляют собой циклоиду, у которой при четных значениях t_1 (0, 2, 4, 6с.) скорость V резко меняет свое направление. В таких острых точках графика понятие производной не имеет смысла, поэтому тангенциальное ускорение $AT=dV/dt$ имеет в них разрыв, что и видно на соответствующей кривой на рис. 3.3.

Отметим, что студентом выполняются только следующие действия:

- записываются нужные уравнения движения материальной точки;
- составляется программа 3.1M на REDUCE;
- программа 3.1M записывается в файле с идентификатором, например, \KIN\PR3-1M с использованием любого текстового редактора;
- файл \KIN\PR3-1M запускается на выполнение в REDUCE, в результате которого получается символьное решение (3.13)–(3.15);
- программа 3.1M дополняется для получения численного решения: в операторах 41, 61 и 86 указывается значение времени, заданное в условии задачи $T_1=1/3$ с (см. дополнение 3.1);
- файл с дополненной программой запускается на выполнение в REDUCE, в результате которого получается численное решение (3.16);

Одновременно с этим обычными методами механики студентом выполняется параллельное аналитическое (безмашинное) решение указанных заданий при одном заданном времени T_1 .

Далее производится сравнение аналитического решения с численным (3.16) при времени T_1 , после совпадения которых убеждаемся в правильности решения.

3.5 ОПРЕДЕЛЕНИЕ КИНЕМАТИЧЕСКИХ ХАРАКТЕРИСТИК ПРИ СЛОЖНОМ ДВИЖЕНИИ ТОЧКИ

Для символического решения на ПК К-7 [23, с. 99-106] (или К-10 [22, с. 137-143]) составление уравнений движения является дополнительной работой, так как без ПК решение заданий на сложное движение точки проводится другим способом с использованием кинематической теоремы Кориолиса. Поэтому рассмотрим их получение для типового примера этого задания ([23, с. 99, 104-106] или [22, с. 137, 141-143]), схема которого представлена на рис. 3.4а.

Исходные данные к представленной на рис. 3.4 схеме типового примера имеют одинаковый вид ([23, с. 99] или [22, с. 137]):

$$\begin{aligned} s_r &= 16 - 8 \cdot \cos(3 \cdot \pi \cdot t) \text{ см;} \\ \varphi_e &= 0,9 \cdot t^2 - 9 \cdot t^3 \text{ рад;} \\ t_1 &= 2/9 \text{ с.} \end{aligned} \tag{3.17}$$

Чтобы составить уравнения движения материальной точки, нужно для произвольного момента времени T выразить ее координаты через данные задачи и параметр T . Время T обычно входит в исходные данные (в уравнения относительного s_r и переносного φ_e движений).

Полученные таким образом выражения координат для одного произвольного положения материальной точки включают переменную T и остаются справедливыми для любого момента времени, являясь уравнениями движения материальной точки в координатном виде в параметрической форме.

Заметим, что в большинстве вариантов задания К-7 [23, с. 101-103] (и К-10 [22, с. 138-140]) исходные схемы следует считать представленными в начальном положении переносного движения при $\varphi_e = 0$.

Поэтому для составления уравнений движения нужно дать произвольное перемещение тела, повернув его *на небольшой угол φ_e по направлению переносного движения* из заданного положения (рис. 3.4а) в произвольное (рис. 3.4б).

В этих заданиях также не определены положения координатных осей и, в зависимости от их выбора, уравнения могут принимать разнообразную форму, отличаясь друг от друга в конечном счете на постоянную. Это

приводит к разным значениям координат точек, определяемых по этим разным уравнениям, но к одинаковым значениям скоростей и ускорений.

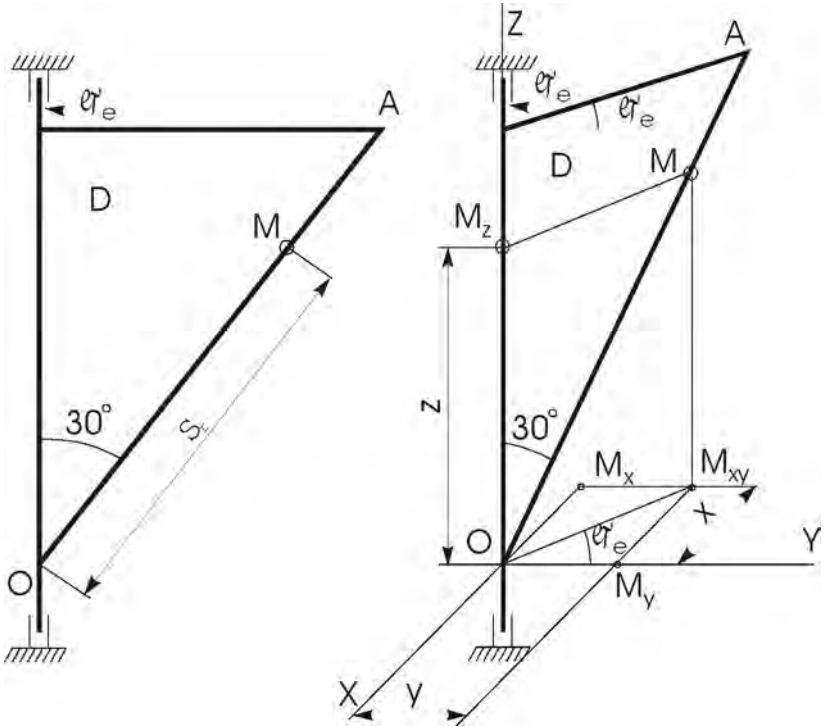


Рис. 3.4. Схема механизма типového примера задания К-7 ([23, с. 99, 104-106] или К-10 [22, с. 137, 141-143]): а — схема механизма; б — дополнения для составления уравнений движения материальной точки М

Выберем координатные оси и переместим тело D из исходного положения на рис. 3.4а в произвольное положение по направлению переносного движения (в сторону положительного отсчета φ_e), как показано на рис. 3.4б. Также условимся, что проекции точек на координатные оси или плоскость будем обозначать той же буквой с индексом данной оси или плоскости: M_x , M_y , M_z , M_{xy} (см. рис. 3.4б). Тогда для произвольного положения точки М ее координаты X, Y и Z через данные задачи и параметр Γ можно выразить следующим образом:

$$\begin{aligned}
X &= -OM_X = -OM_{XY} * \sin \varphi_e = -OM * \sin 30^\circ * \sin \varphi_e = -(s_r/2) * \sin \varphi_e; \\
Y &= OM_Y = OM_{XY} * \cos \varphi_e = OM * \sin 30^\circ * \cos \varphi_e = (s_r/2) * \cos \varphi_e; \quad (3.18) \\
Z &= OM_z = OM * \cos 30^\circ = s_r * \cos 30^\circ.
\end{aligned}$$

Для удобства составления программ заменим также используемые греческие и латинские буквы в обозначениях углов и индексах переменных их идентификаторами, записанными в латинской транскрипции:

SR	FIE	PI	
s_r	φ_e	π	(3.19)

С учетом соотношений (3.19), значений s_r и φ_e (3.17) уравнения движения (3.18) примут следующий вид:

$$\begin{aligned}
X &= -(SR/2) * \sin(FIE), \\
Y &= (SR/2) * \cos(FIE), \\
Z &= SR * \cos(PI/6),
\end{aligned} \quad (3.20)$$

где $SR=16-8*\cos(3*PI*T)$, $FIE=0.9*T**2-9*T**3$, $PI=3.1415926$.

Отметим, что типовые примеры заданий К-7 [23, с. 99, 104-106] и К-10 [22, с. 137, 141-143] одинаковы.

Поэтому их программная реализация совпадает и рассматривается на примере (3.20).

Теперь для решения типового примера задания К-7 в символьном виде в программе 3.1 нужно:

- исправить уравнения движения (операторы 25 и 27) по фрагменту (3.20);
- учесть пространственный характер задачи:
 - дать соответствующие выражения для VZ и AZ ;
 - исправить определения для скорости V и ускорения A , введя в операторы 40 и 60 дополнительные слагаемые $VZ*VZ$ и $AZ*AZ$ соответственно;
 - удалить операторы 70-85 из-за отсутствия необходимости при сложном движении точки определять радиус кривизны траектории.

Хотя изложенных рекомендаций достаточно для самостоятельного составления программы, для большей ясности изложения мы приведем получающийся результат полностью в нижеследующей программе 3.2.

COMMENT **Программа 3.2:** Составление уравнений движения при сложном движении точки и определение ее скорости и ускорения для типового примера К-7 [23, с. 99, 104-106] (или К-10 [22, с. 137, 141-143]) на REDUCE с представлением результатов в символьном виде;

```

OUT "\KIN\PR3-2.LIS";                                10
ON FORT;                                              15
CLEAR T1;                                           20
SR:=16-8*COS(3*PI*T1);                               23
FIE:=0.9*T1**2-9*T1**3;                             24
X:=- (SR/2)*SIN(FIE);                               25
Y:=(SR/2)*COS(FIE);                                 27
Z:=SR*COS(PI/6);                                    29
VX:=DF(X, T1);                                      30
VY:=DF(Y, T1);                                      32
VZ:=DF(Z, T1);                                      34
V:=SQRT(VX*VX+VY*VY+VZ*VZ);                        40
AX:=DF(VX, T1);                                     50
AY:=DF(VY, T1);                                     52
AZ:=DF(VZ, T1);                                     54
A:=SQRT(AX*AX+AY*AY+AZ*AZ);                        60
SHUT "\KIN\PR3-2.LIS";                              90
OFF FORT;                                            95
END;                                                 99

```

Команда 10 открывает, а команда 90 закрывает выводной файл PR3-2.LIS, находящийся в подкаталоге \KIN корневого каталога текущего диска, куда записываются результаты работы программы 3.2.

Напомним, что здесь, как и ранее предполагается, что программа 3.2 расположена в том же текущем подкаталоге \KIN в файле, например, с именем PR3-2.

Команда 15 поднимает, а команда 90 опускает флаг FORT, устанавливающий выдачу результатов в форме, совместимой с Фортраном. Это сделано как для удобства представления предполагаемых сложных и длинных результатов расчета, так и для показа возможности проведения дальнейших сложных численных расчетов с использованием фортран-программ.

Команда CLEAR 20 для большей надежности очищает переменную T1 перед первым использованием в программе, делая ее свободной, так что ее значением является сам символ T1.

Операторы 23 и 24 задают значения вспомогательных переменных SR и FIE с учетом данных условия (3.17).

Операторы 25, 27 и 29 присваивают алгебраические выражения, описывающие уравнения движения материальной точки в виде зависимости ее проекций от времени T1, переменным X, Y и Z.

Операторы 30, 32 и 34 определяют проекции скорости точки путем дифференцирования алгебраических выражений для соответствующих координат X, Y и Z по времени T1. Здесь число, выражающее порядок производной, равно единице, поэтому оно опущено и не указывается в качестве третьего параметра.

Оператор 40 находит значение модуля скорости точки V путем извлечения квадратного корня из сумм квадратов проекций скорости VX, VY и VZ.

Операторы 50, 52 и 54 определяют проекции ускорения точки путем взятия первой производной от алгебраических выражений для соответствующих проекций скорости точки VX, VY и VZ по времени T1. Здесь порядок производной, равный единице, также не указывается.

Операторы 50, 52 и 54 можно представить также в форме второй производной от соответствующих координат X, Y и Z, где порядок следует указывать обязательно. Это может иметь, например, следующий вид:

$$\begin{aligned} AX &:= DF(X, T1, 2); & 50 \\ AY &:= DF(Y, T1, 2); & 52 \\ AZ &:= DF(Z, T1, 2); & 54 \end{aligned}$$

Оператор 60 находит значение ускорения точки A путем извлечения квадратного корня из сумм квадратов проекций ускорений AX, AY и AZ.

Оператор 99, как обычно, запускает специальный учет контроля за файлами, улучшающий эффективность работы системы, что является

стандартным способом заканчивать файлы, предназначенные для считывания.

Теперь следует записать на диске набранную программу 3.2 в файл, например, с именем \KIN\PR3-2. После чего для запуска ее на выполнение нужно выйти из рабочего подкаталога \KIN корневого каталога текущего диска, перейти в подкаталог CAB REDUCE с расположенными в нем системными файлами и загрузить систему. Затем после появления приглашения в командной строке следует ввести команду IN в форме (3.5) или (3.7) с указанием имени запускаемого на выполнение файла, например:

1: in "\KIN\PR3-2" S (3.21)

По этой команде в систему будет загружен файл \KIN\PR3-2, выполнены все находящиеся в нем операторы и команды программы 3.2, а результаты их выполнения будут записаны на диске в файле PR3-2.LIS в том же подкаталоге \KIN корневого каталога текущего диска. Они будут иметь для рассматриваемого примера (3.20) следующие значения. В целях экономии места приведем только получающиеся аналитические зависимости для:

- скорости V

$$\begin{aligned} \text{ANS4} = & -270. * \text{COS}(3. * \text{PI} * \text{T1}) ** 2 * \text{SIN}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / \\ & 10.) ** 2 * \text{T1} ** 3 + 9. * \text{COS}(3. * \text{PI} * \text{T1}) ** 2 * \text{SIN}((90. * \text{T1} ** 3 - 9. * \\ & \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 2 - 8100. * \text{COS}(3. * \text{PI} * \text{T1}) * \text{SIN}((90. * \text{T1} \\ & ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 4 + 1080. * \text{COS}(3. * \text{PI} * \text{T1}) * \text{SIN} \\ & ((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 3 - 36. * \text{COS}(3. * \text{PI} * \text{T1}) \\ & * \text{SIN}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 2 + 25. * \text{SIN}((90. \\ & * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{SIN}(3. * \text{PI} * \text{T1}) ** 2 * \text{PI} ** 2 + \\ & 8100 * \text{SIN}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 4 - 1080. * \\ & \text{SIN}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 3 + 36. * \text{SIN}((90. * \\ & \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{T1} ** 2 + 75. * \text{SIN}(3. * \text{PI} * \text{T1}) ** 2 * \\ & \text{PI} ** 2 \end{aligned} \quad (3.22)$$

$$\begin{aligned} \text{ANS3} = & 2025. * \text{COS}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \text{COS}(3. * \text{PI} \\ & * \text{T1}) ** 2 * \text{T1} ** 4 - 270. * \text{COS}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / 10.) ** 2 * \\ & \text{COS}(3. * \text{PI} * \text{T1}) ** 2 * \text{T1} ** 3 + 9. * \text{COS}((90. * \text{T1} ** 3 - 9. * \text{T1} ** 2) / \end{aligned}$$

```

. 10.)**2*COS(3.*PI*T1)**2*T1**2-8100.*COS((90.*T1**3-
. 9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**4+1080.*COS((90.*
. T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**3-36.*COS(
. (90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**2+25.
. *COS((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)**2*
. PI**2+8100.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1**4-
. 1080.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1**3+36.*COS(
. (90.*T1**3-9.*T1**2)/10.)**2*T1**2+2025.*COS(3.*PI*
. T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**4+ANS4
ANS2=SQRT(ANS3)
ANS1=12.*ANS2
V=ANS1/5.

```

• ускорения А

```

ANS10=437400.*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**6-
. 19440.*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**5+324.*
. SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**4+90000.*SIN((
. 90.*T1**3-9.*T1**2)/10.)**2*T1**2-6000.*SIN((90.*T1**
. 3-9.*T1**2)/10.)**2*T1+100.*SIN((90.*T1**3-9.*T1**2)
. /10.)**2
ANS9=-90000.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*T1**2)/
. 10.)**2*T1**2+6000.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*
. T1**2)/10.)**2*T1-100.*COS(3.*PI*T1)*SIN((90.*T1**3-
. 9.*T1**2)/10.)**2+202500.*SIN((90.*T1**3-9.*T1**2)/
. 10.)**2*SIN(3.*PI*T1)**2*PI**2*T1**4-27000.*SIN((90.*
. T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)**2*PI**2*T1**3
. +900.*SIN((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)
. **2*PI**2*T1**2+270000.*SIN((90.*T1**3-9.*T1**2)/10.
. )**2*SIN(3.*PI*T1)*PI*T1**3-27000.*SIN((90.*T1**3-9.
. *T1**2)/10.)**2*SIN(3.*PI*T1)*PI*T1**2+600.*SIN((90.

```



```

. *T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)*PI*T1+
. 16402500.*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**8-
. 4374000.*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**7+ANS10
ANS8=13500.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*T1**2)/
. 10.)**2*SIN(3.*PI*T1)*PI*T1**2-300.*COS(3.*PI*T1)*SIN
. ((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)*PI*T1-
. 202500.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*T1**2)/10.)**
. 2*PI**2*T1**4+27000.*COS(3.*PI*T1)*SIN((90.*T1**3-9.
. *T1**2)/10.)**2*PI**2*T1**3-900.*COS(3.*PI*T1)*SIN((
. 90.*T1**3-9.*T1**2)/10.)**2*PI**2*T1**2-16402500.*COS
. (3.*PI*T1)*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**8+
. 4374000.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*T1**2)/10.)
. **2*T1**7-437400.*COS(3.*PI*T1)*SIN((90.*T1**3-9.*T1
. **2)/10.)**2*T1**6+19440.*COS(3.*PI*T1)*SIN((90.*T1
. **3-9.*T1**2)/10.)**2*T1**5-324.*COS(3.*PI*T1)*SIN((
. 90.*T1**3-9.*T1**2)/10.)**2*T1**4+ANS9
ANS7=450.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/
. 10.)**2*PI**2*T1**2+4100625.*COS(3.*PI*T1)**2*SIN((
. 90.*T1**3-9.*T1**2)/10.)**2*T1**8-1093500.*COS(3.*PI*
. T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**7+
. 109350.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/10.
. )**2*T1**6-4860.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*
. T1**2)/10.)**2*T1**5+81.*COS(3.*PI*T1)**2*SIN((90.*
. T1**3-9.*T1**2)/10.)**2*T1**4+22500.*COS(3.*PI*T1)**
. 2*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**2-1500.*COS(
. 3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1+25.*
. COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2+
. 1875.*COS(3.*PI*T1)**2*PI**4-135000.*COS(3.*PI*T1)*
. SIN((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)*PI*T1
. **3+ANS8

```

```

ANS6=600.*COS((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*
. T1)*PI*T1+16402500.*COS((90.*T1**3-9.*T1**2)/10.)**2
. *T1**8-4374000.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1
. **7+437400.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1**6-
. 19440.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1**5+324.*
. COS((90.*T1**3-9.*T1**2)/10.)**2*T1**4+90000.*COS((
. 90.*T1**3-9.*T1**2)/10.)**2*T1**2-6000.*COS((90.*T1**
. 3-9.*T1**2)/10.)**2*T1+100.*COS((90.*T1**3-9.*T1**2)
. /10.)**2+625.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1
. **2)/10.)**2*PI**4+101250.*COS(3.*PI*T1)**2*SIN((90.
. *T1**3-9.*T1**2)/10.)**2*PI**2*T1**4-13500.*COS(3.*
. PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2*PI**2*T1
. **3+ANS7 (3.23)
ANS5=-437400.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.
. *PI*T1)*T1**6+19440.*COS((90.*T1**3-9.*T1**2)/10.)**
. 2*COS(3.*PI*T1)*T1**5-324.*COS((90.*T1**3-9.*T1**2)/
. 10.)**2*COS(3.*PI*T1)*T1**4-90000.*COS((90.*T1**3-9.*
. T1**2)/10.)**2*COS(3.*PI*T1)*T1**2+6000.*COS((90.*T1
. **3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1-100.*COS((90.
. *T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)+202500.*COS((
. 90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)**2*PI**2*T1
. **4-27000.*COS((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*
. PI*T1)**2*PI**2*T1**3+900.*COS((90.*T1**3-9.*T1**2)/
. 10.)**2*SIN(3.*PI*T1)**2*PI**2*T1**2+270000.*COS((90.
. *T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)*PI*T1**3-
. 27000.*COS((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)
. *PI*T1**2+ANS6
ANS4=-1500.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*
. PI*T1)**2*T1+25.*COS((90.*T1**3-9.*T1**2)/10.)**2*

```

```

. COS(3.*PI*T1)**2-135000.*COS((90.*T1**3-9.*T1**2)/
. 10.)**2*COS(3.*PI*T1)*SIN(3.*PI*T1)*PI*T1**3+13500.*
. COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*SIN(
. 3.*PI*T1)*PI*T1**2-300.*COS((90.*T1**3-9.*T1**2)/10.)
. **2*COS(3.*PI*T1)*SIN(3.*PI*T1)*PI*T1-202500.*COS((
. 90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*PI**2*T1**4
. +27000.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*
. T1)*PI**2*T1**3-900.*COS((90.*T1**3-9.*T1**2)/10.)**
. 2*COS(3.*PI*T1)*PI**2*T1**2-16402500.*COS((90.*T1**3
. -9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**8+4374000.*COS(
. (90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**7+
. ANS5
ANS3=625.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*
. T1)**2*PI**4+101250.*COS((90.*T1**3-9.*T1**2)/10.)**
. 2*COS(3.*PI*T1)**2*PI**2*T1**4-13500.*COS((90.*T1**3
. -9.*T1**2)/10.)**2*COS(3.*PI*T1)**2*PI**2*T1**3+450.
. *COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)**2*
. PI**2*T1**2+4100625.*COS((90.*T1**3-9.*T1**2)/10.)**
. 2*COS(3.*PI*T1)**2*T1**8-1093500.*COS((90.*T1**3-9.*
. T1**2)/10.)**2*COS(3.*PI*T1)**2*T1**7+109350.*COS((
. 90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)**2*T1**6-
. 4860.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)
. **2*T1**5+81.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(
. 3.*PI*T1)**2*T1**4+22500.*COS((90.*T1**3-9.*T1**2)/
. 10.)**2*COS(3.*PI*T1)**2*T1**2+ANS4
ANS2=SQRT(ANS3)
ANS1=36.*ANS2
A=ANS1/25.

```

Отметим, что именно для выработки бесстрашия перед такими огромными алгебраическими выражениями, ужасающими для ручной обработки, они здесь и приведены.

Получены они компьютером, им же будут и дальше обрабатываться. Ваше дело — лишь вовремя давать ему указания путём набора весьма простых команд и, после получения аналитического решения и его графической интерпретации, сосредоточиться на его осмыслении и анализе.

Режим FORT включает все достоинства выключения флага NAT при получении длинных результатов для представления их по строкам. Кроме этого строго соблюдается синтаксис фортран-программ:

- все выражения начинаются с 7-й колонки;
- если запись не помещается на одной строке, то в последующих строках появится признак продолжения — знак "." (точка) в 6-й колонке, сопровождаемая пробелом (максимальное количество строк продолжения 19);
- при записи длинных выражений используются вспомогательные переменные (ANS4–ANS1 для скорости V в (3.22) и ANS10–ANS1 для ускорения A в (3.23)). Они разбивают громоздкое выражение на ряд более простых. Их число, как и остальные режимы вывода, могут быть по желанию изменены (см. дополнение 3.2). Имя ANS с соответствующим номером присваивается каждому выражению по умолчанию и все его строки также печатаются с седьмой позиции с использованием синтаксиса Фортрана.

Теперь дополним полученное символьное решение (3.22)–(3.23) соответствующим численным при конкретном значении времени T_1 . Для этого следует в программе 3.2, дополненной по фрагменту дополнения 3.1 без команд 86–89, в операторах 41 и 61 указать значение времени, заданное в условии задачи: $T_1=2/9$ с.

После выполнения такой дополненной программы 3.2 по команде (3.21), результаты численного решения для скорости точки V и ускорения точки A для рассматриваемого типового примера будут иметь соответственно следующие значения:

$$V = 65.9604443, A = 394.9225358. \quad (3.24)$$

Сравнив их с результатами вашего аналитического (безмашинного) расчета при одинаковом значении времени T_1 убеждаемся в правильности решения. После чего строим по соотношениям (3.22)–(3.23) на ПК с ис-

пользованием системы DERIVE (см. пособия [6, 14]) графики, представленные на рис. 3.5.

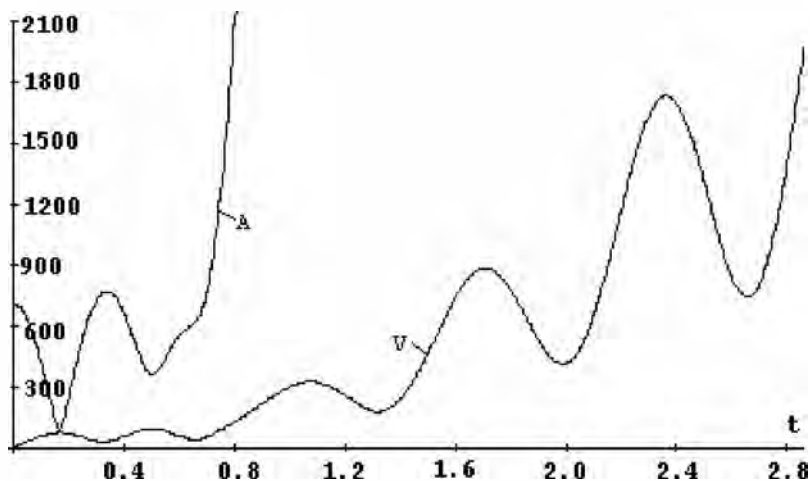


Рис. 3.5. Зависимости скорости V (м/с) и полного ускорения A (м/с²) от времени t (с) при сложном движении материальной точки для типового примера заданий К-7 [23, с. 99, 104–106] или К-10 [22, с. 137, 141–143].

Отметим, что пользователем также составляются уравнения движения материальной точки при ее сложном движении и программа на REDUCE для символьного определения кинематических характеристик при различных временах T (что называется символьным решением). Затем следует сеанс работы на ПК и получение символьного (3.22)–(3.23) и численного (3.24) решения. Одновременно с этим обычными методами механики студентом производится параллельное аналитическое (безмашинное) решение указанных заданий при одном заданном времени t_1 (которое рассмотрено в сборнике [23, с. 99, 104–106] или К-10 [22, с. 137, 141–143] и в пособии не повторяется). После его совпадения с численным решением (3.24), убедившись в правильности решения, на ПК с использованием системы DERIVE (см. [6, 14]) строятся графики.

При этом не следует пугаться внешнего вида формул типа (3.22)–(3.23). Вам не придется их проверять и набирать вручную для ввода в САВ DERIVE. Они будут считаны в буфер обмена, быстро и безошибочно вставлены в текстовый редактор Word, где машинным образом некоторые специальные символы будут представлены в понимаемой системой DE-

RIVE форме, опять скопированы в буфер и вставлены в CAB DERIVE, после чего эта система быстро и точно построит нужные графики. Все будет сделано действительно с легким сердцем и удовольствием, которые доставляют современные CAB.

Дополнение 3.2. Имеется несколько способов, позволяющих изменить автоматический формат вывода в фортрановском виде.

1. Задание с помощью оператора присваивания соответствующего значения внутренней редьюсовской переменной CARDNO*

CARDNO!* := число; (3.25)

определит количество строк продолжения для записи одного выражения. От заданного значения зависит число вспомогательных переменных, которые используются для записи сложных выражений. В начальном состоянии параметр CARDNO* устанавливается равным 20. При этом значении для скорости V в (3.22) используются вспомогательные переменные ANS4–ANS1 и ANS10–ANS1 для ускорения A в (3.23).

Их использование, как обычно считается, удобно для записи программы на Фортране, ибо они разбивают громоздкое выражение на ряд более простых. Но при машинной генерации выражений в аналитическом виде, записи их во внешний файл и последующей также компьютерной обработке, это достоинство не несет практически никаких выгод, ибо простота этих выражений относительна (см. (3.22) и (3.23)).

Кроме этого, использование вспомогательных переменных приводит к тому, что мы не можем передать все сложное выражение за один раз через буфер обмена (<Ctrl>+C – <Ctrl>+V), а вынуждены считывать его порциями по значениям каждой вспомогательной переменной. При этом емкость буфера обмена не накладывает практически никаких ограничений. Поэтому сложность и громоздкость передаваемого выражения не имеют никакого значения. Важно, чтобы оно могло быть считано в буфер обмена за один раз.

Для достижения этой цели достаточно только соответствующим образом изменить значение переменной CARDNO*. Например, для получения выражений (3.22) и (3.23) без использования вспомогательных переменных следует задать значение параметра CARDNO* соответственно более 34 для скорости V (см. оператор № фрагмента программы дополнение 3.3) и 145 для ускорения A.

Отметим, что для включения в состав имен переменных (кроме латинских букв и цифр) разрешенных символов (например, * у переменной CARDNO*), *перед ним* всегда указывается специальный символ, которым может быть в зависимости от версии REDUCE восклицательный знак (“!”) или закрывающая скобка (“]”).

2. Задание с помощью оператора присваивания соответствующего значения внутренней редьюсовской переменной FORTWIDTH*

```
FORTWIDTH!* := число; (3.26)
```

определит количество столбцов при выводе сложного выражения. В исходном состоянии системы это число равно 70.

3. Система REDUCE после каждого изолированного целочисленного коэффициента в фортрановском выражении автоматически вставляет десятичную точку (таким образом, что 4 становится 4.). Для предотвращения этого следует опустить флаг PERIOD

```
OFF PERIOD; (3.27)
```

4. Имя ANS, автоматически присвоенное непоименованному переменному выражению и его частям, может быть изменено с помощью операции VARNAME.

Эта операция имеет в качестве параметра один идентификатор, который и заменяет идентификатор ANS в качестве имени выражения. Значением операции VARNAME является её параметр. Например:

```
VARNAME PEREM; 17
```

Теперь, после включения предложения 17 в программу 3.2, в результате ее работы аналитические ее выражения (3.22) и (3.23) будут представлены с использованием вспомогательных переменных PEREM4 – PEREM1 для скорости V и PEREM10 – PEREM1 для ускорения A (вместо соответственно ANS4 – ANS1 в (3.22) и ANS10 – ANS1 в (3.23)).

Дополнение 3.3. Приведем пример записи программ в синтаксисе Фортрана во внешний файл.

Эта возможность организована при помощи флага FORT с использованием оператора WRITE. Возьмем в качестве основы программу 3.2. Ограничимся с целью экономии места дополнением для представления в синтаксисе Фортрана только скорости точки V, которое получим без использования вспомогательных переменных, задав соответствующим образом значение CARDNO*. Для этого в программу 3.2 нужно добавить следующий фрагмент:

CARDNO!*:=34\$	41
OUT "\PR\D3-2.FOR";	42
WRITE " PI=3.141592654"\$	43
WRITE " T1=2./9."\$	44
V;	45
WRITE " PRINT 20, ANS"\$	46
WRITE "20 FORMAT(5X, 'СКОРОСТЬ V = ',G12.5)"\$	47
WRITE " STOP"\$	48
WRITE " END"\$	49
SHUT "\PR\D3-2.FOR"\$	49a
OUT "\KIN\PR3-2.LIS";	49б

Оператор 41 задает значение внутренней редьюсовской переменной CARDNO*, которая определяет количество строк продолжения для записи одного выражения. Оно предварительно подобрано таким образом, чтобы вызвать печать значения для скорости V без использования вспомогательных переменных.

Команда 42 открывает второй выводной файл D3-2.FOR, находящийся в подкаталоге \PR корневого каталога текущего диска, куда будут записываться результаты работы операторов 43–49 программы 3.2 с дополнением 3.3. Напомним, что сама программа 3.2 расположена в подкаталоге \KIN в файле с именем PR3-2. Там же расположен и первый выводной файл PR3-2.LIS, открытый командой 10. В него записывались результаты работы операторов 22-40 программы 3.2.

В командах WRITE 43–44 и 46–49, в качестве списка параметров используется текст вспомогательных операторов Фортрана, заключённый в кавычки. Он выводится в том же виде без кавычек на печать. Для записи операторов Фортрана с 7-й позиции и для их отделения друг от друга в кавычках указываются пробелы.

Команды 43–44 печатают арифметические операторы присваивания Фортрана, задающие значения переменным PI и T1. Отметим, что в REDUCE переменная PI является зарезервированной и попытка присваивания ей значения приведет к ошибке, в то время как для Фортрана задание ее значения обязательно.

Команда 45 производит печать выражения для скорости точки V, определяемого оператором 40 программы 3.2. Вывод производится в синтаксисе Фортрана с учетом значения внутренней редьюсовской перемен-

ной CARDNO*. Так как выражение только распечатывается, а не определяется, как в операторе 40, то оно присваивается по умолчанию переменной ANS.

Команды 46 и 47 печатают операторы PRINT и FORMAT, выполняющих печать значения переменной ANS, вычисленной после работы программы на Фортране, впереди которого будет напечатан заголовок “СКОРОСТЬ V = ”.

Команды 48 и 47 печатают операторы Фортрана STOP и END, указывающие соответственно на конец вычислений и конец основной программы.

Команда 49а закрывает второй выводной файл D3-2.FOR, находящийся в подкаталоге \PR корневого каталога текущего диска, куда записывались результаты работы операторов 43–49 программы 3.2 с дополнением 3.3.

Команда 49б повторно открывает первый выводной файл PR3-2.LIS, расположенный в подкаталоге \KIN. Если этого не сделать, то после закрытия командой 49а выводного файла D3-2.FOR, вывод будет направляться на терминал.

После второй команды 10 под номером 49б вывод будет происходить в конец файла PR3-2.LIS. Теперь результаты работы операторов 50-60 также будут записаны в этом файле. Поэтому основной вывод программы 3.2 не изменится после добавления операторов 41-49б дополнения 3.3 (кроме числа вспомогательных переменных для печати ускорения A вследствие изменения оператором 41 значения внутренней редьюсовской переменной CARDNO*).

После запуска на выполнение файла PR3-2 с дополнением операторов 41-49б по команде (3.21), в файле D3-2.FOR, находящемся в подкаталоге PR корневого каталога диска C, будет находиться исходный текст программы на Фортране для численного определения скорости V по полученным в REDUCE выражениям, приведенный ниже:

```
PI=3.141592654
T1=2./9.
ANS=(12.*SQRT(2025.*COS((90.*T1**3-9.*T1**2)/10.))**2
. COS(3.*PI*T1)**2*T1**4-270.*COS((90.*T1**3-9.*T1**2)
. /10.))**2*COS(3.*PI*T1)**2*T1**3+9.*COS((90.*T1**3-9.
. *T1**2)/10.))**2*COS(3.*PI*T1)**2*T1**2-8100.*COS((
```



```

. 90.*T1**3-
9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**4+1080.
. *COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)*T1**
. 3-36.*COS((90.*T1**3-9.*T1**2)/10.)**2*COS(3.*PI*T1)
. *T1**2+25.*COS((90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*
. PI*T1)**2*PI**2+8100.*COS((90.*T1**3-9.*T1**2)/10.)
. **2*T1**4-1080.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1
. **3+36.*COS((90.*T1**3-9.*T1**2)/10.)**2*T1**2+2025.
. *COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)/10.)**2*
. T1**4-270.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.*T1**2)
. /10.)**2*T1**3+9.*COS(3.*PI*T1)**2*SIN((90.*T1**3-9.
. *T1**2)/10.)**2*T1**2-8100.*COS(3.*PI*T1)*SIN((90.*
. T1**3-9.*T1**2)/10.)**2*T1**4+1080.*COS(3.*PI*T1)*
. SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**3-36.*COS(3.*PI
. *T1)*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**2+25.*SIN(
. (90.*T1**3-9.*T1**2)/10.)**2*SIN(3.*PI*T1)**2*PI**2+
. 8100.*SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**4-1080.*
. SIN((90.*T1**3-9.*T1**2)/10.)**2*T1**3+36.*SIN((90.*
. T1**3-9.*T1**2)/10.)**2*T1**2+75.*SIN(3.*PI*T1)**2*
. PI**2))/5. (3.28)

```

```

20  FORMAT(5X, 'СКОРОСТЬ V = ',G12.5)
    PRINT 20, ANS
    STOP
    END

```

Дополнение 3.4. Рассмотрим для удобства пользования пособием дальнейшую обработку полученной программы (3.28) на Фортране: последующую ее трансляцию и выполнение на ПК.

Ввод ее исходного текста (в файл D3-2.FOR) был осуществлен в результате работы модифицированной по дополнению 3.3 программы 3.2 на REDUCE. Для получения выполняемой программы (с расширением .exe) нужно произвести ее компиляцию и редактирование. Вызов компилятора (транслятора) производится по команде FL, в которой указываются режимы компиляции, начинающиеся с символа “/” (наклонная черта), и имя обрабатываемого файла. После ее окончания создается объектный файл (тип .obj). Сообщения об ошибках в программе выводятся на экран и помещаются также в отдельный файл (тип .lst).

Если в программе отсутствуют серьезные ошибки, то с помощью команды LINK вызывается Редактор связей для создания выполняемого файла (тип .exe). В процессе редактирования к основному объектному файлу присоединяются необходимые программы из библиотеки Фортрана, а также объектные файлы, дополнительно указанные в команде.

Команда FL имеет следующий формат:

```
FL [режим] [имя-файла] [режим] [имя-файла]...
```

```
[/link [библиотеки] [режимы-редактирования]] (3.29)
```

где режим – один из режимов компилятора (см. п. 8.4.1 [17, с. 141-143]);

имя-файла – идентификатор исходного файла;

библиотеки – список имен библиотечных файлов, явно заданных для редактирования;

режимы-редактирования - режимы Редактора связей (см. п. 8.4.2 [17, с. 143-145]).

Если режим /link используется в команде FL, то компилятор автоматически вызывает Редактор связей для создания выполняемого файла, если нет – то потребуются отдельный шаг редактирования с помощью команды LINK (см. п. 8.4 [17, с. 140-145]). Команды FL и LINK вследствие своей сложности помещаются обычно в командные файлы (с типом .bat).

Отметим, что режимы могут быть записаны в любом месте команды FL. Они относятся ко всем файлам, имена которых следуют за ними в команде. Не смотря на то, что имена файлов и команд могут задаваться как строчными, так и прописными буквами, режимы в команде FL следует задавать так, как они указаны в пособии, например, /FpI.

При указании режима /HELP (или /help) выдается справочная информация о команде FL. Например, по команде FL /HELP она выводится на экран, а по FL /HELP >PRN перенаправляется на печатающее устройство.

Пусть исполняемые файлы компилятора Фортрана помещены в подкаталоге C:\F5\BIN, библиотечные - в C:\F5\LIB, графические - в C:\F5\INCLUDE, а временно создаваемые файлы в подкаталоге C:\F5\TMP. Создание этих подкаталогов и помещение в них файлов выполняется автоматически после указания их имен в ответ на соответствующие вопросы при инсталляции Фортрана. Будем также считать, что к ним указан путь в команде PATH файла AUTOEXEC.BAT:

```
PATH C:\F5\BIN;C:\F5\LIB;C:\F5\INCLUDE;C:\F5\TMP (3.30)
```

Отметим, что в списке каталогов, задаваемых командой PATH, обычно перечисляются через точку с запятой те каталоги данного ПК, в которых находятся исполняемые программы общего назначения. Поэтому не следует обращать внимание на их список.

Важно только, чтобы в их числе находились пути команды (3.30). Рекомендуется также имена каталогов в команде PATH указывать полностью от корневого каталога соответствующего диска (как указано в (3.30)), что позволит командному процессору DOS правильно их находить из любого текущего каталога и диска.

В подкаталоге C:\F5\BIN создадим командный файл FFL.BAT для запуска компилятора Фортрана (что позволит вызывать его из любого текущего подкаталога):

```
@ECHO OFF 10
PATH C:\F5\BIN; 20
SET LIB=C:\F5\LIB 30
SET INCLUDE=C:\F5\INCLUDE 40 (3.31)
SET TMP=C:\F5\TMP 50
FL /Fs /Fpi %1.FOR 60
```

Команда 10 предназначена для того, чтобы последующие не выводились на экран.

Команда 20 указывает путь в подкаталог, в котором находятся исполняемые файлы компилятора.

Команды 30-50 задают переменные окружения, указывающие расположение подкаталогов, в которых находятся библиотечные, графические и временно создаваемые файлы.

Команда 60 производит вызов программы компилятора Фортрана. Режим /Fs используется для вывода распечатки исходного текста компилируемых модулей программ (с типом .LST), которая полезна при их отладке.

По режиму /с выполняется только компиляция исходных файлов, заданных в команде. По умолчанию объектному файлу (с типом .OBJ) присваивается такое же основное имя, как у соответствующего исходного файла. Если режим /с в команде FL отсутствует, то используется по умолчанию режим /link, полученный объектный файл редактируется и создается выполняемый файл (с типом .EXE). Это удобно при работе с

программами, все модули которых располагаются в одном файле, как в нашем случае. При компиляции подпрограмм, находящихся в отдельных файлах, следует использовать режим /с (см. п. 8.4 [17, с. 140-145]).

Согласование режима /FP, осуществляющего управление операциями с плавающей точкой, и используемой библиотеки Фортрана является особенно важным (см. [17, с. 140-145]). Режим /FPi полезен, когда нет уверенности в наличии математического сопроцессора. Если он присутствует, то программа его использует. При отсутствии сопроцессора режим /FPi обеспечивает получение наиболее высокой точности выполнения операций с плавающей точкой.

Имя компилируемой программы, указываемое при запуске после имени командного файла (FFLL в нашем случае), является заменяемым параметром. Оно будет подставлено при выполнении команд вместо символа %1. Тип файла (.FOR) уже указан в процедуре (3.31), поэтому повторно его указывать не нужно. Например, для компиляции программы (3.28), находящейся в файле D3-2.FOR, следует из соответствующего текущего подкаталога ввести команду:

```
FFLL D3-2 (3.32)
```

При запуске выполняемого файла (с расширением .EXE) во всех случаях следует учитывать рекомендации п. 7.6.3 [17, с. 89-90]:

- предварительно использовать в программах описанный там же оператор OPEN;
- применять перенаправление потоков ввода-вывода (см. п. 2.2.3 [17, с. 26]).

В последнем случае нужно указать посредством использования в команде знака “меньше” (<) соответствующее имя файла, из которого следует брать исходные данные. Для указания того, куда помещать результаты расчета, используется знак “больше” (>).

Исходные данные в нашем случае задаются в самой программе, поэтому для выполняемого файла D3-2.EXE это может иметь вид:

```
D3-2.EXE > D3-2.LIS (3.33)
```

После запуска команды (3.33) на выполнение в том же текущем подкаталоге PR корневого каталога диска C будет находиться файл D3-2.LIS, в котором будут находиться результаты работы фортран-программы (3.28), приведенные ниже:

СКОРОСТЬ $v = 65.960$

Сравнив их с полученными ранее на REDUCE результатами (3.24) для скорости точки, убеждаемся в правильности решения.

В заключение отметим, что в работе [9] содержится большое количество процедур, а в пособии [24] – программ, которые удобно использовать в различных задачах механики и математики. Если при этом вы встретите трудности, почувствовав недостаток теоретического багажа, то в монографии [7] наиболее подробно описаны возможности операторов и команд базовой версии REDUCE.

ЧАСТЬ 2. КРАТКИЕ СВЕДЕНИЯ ДЛЯ ПРАКТИЧЕСКОЙ РАБОТЫ НА ПК

4. ОБЩЕЕ ОПИСАНИЕ ПК

Приведем необходимые сведения, достаточные для самостоятельной работы на IBM-совместимых ПК.

4.1 СРАВНИТЕЛЬНОЕ ОПИСАНИЕ КЛАВИАТУР ЗАРУБЕЖНЫХ И ОТЕЧЕСТВЕННЫХ ПК

1. В центральном поле клавиатуры находятся алфавитно-цифровые и знаковые клавиши, расположенные примерно как на пишущей машинке.

2. Верхний ряд (над центральным полем) содержит функциональные клавиши <F1>–<F12>, обычно разделенные на три группы по 4 клавиши в каждой (в некоторых ПК они находятся в левой части и содержат 10 клавиш <F1>–<F10>). На компьютерах отечественного производства они обозначаются <Ф1>–<Ф12> или <Ф1>–<Ф10> соответственно. Поскольку каждая функциональная клавиша при нажатии передает уникальный (непечатаемый) код, прикладные программы могут присваивать специальные значения этим клавишам. После этого можно одним нажатием клавиши ввести в ЭВМ часто используемую сложную командную последовательность (дисковой операционной системы — DOS) или последовательность обычно используемых операций. Это сокращает объем и время работы, поскольку нажатием одной клавиши можно заменить необходимость громоздкого набора этих команд с клавиатуры. Отметим, что функциональные клавиши могут выполнять разные задачи для различных прикладных программ, поскольку не существует стандартов, определяющих присвоение программных функций каждой клавише.

3. Малое поле в правой части клавиатуры, где расположены цифровые и управляющие клавиши, образуют двухрежимную малую цифровую клавиатуру. При включенном режиме NumLock (Цифровая блокировка)

эта самая правая группа клавиш дублирует цифровые клавиши, а при выключенном — клавиши управления перемещения курсора.

4. Слева и справа центральное светлое поле обрамляют стандартные управляющие клавиши. Рассмотрим их функциональное назначение и соответствие для клавиатур ПК зарубежного и отечественного производства, которые будем разделять тире “—”. Основные модификации названий для каждой из клавиатур указываются через запятую.

<Enter>, <CR>, <Return> – <ВВОД>, <↵> — ввод текущей строки и перевод курсора в начало следующей строки. ввод каждой команды также должен оканчиваться нажатием клавиши <Enter> – <ВВОД>.

<Esc> – <КЛЮЧ>, <Спец>, <ВЫХ> — сокращенное написание слова “Escape” (“Переход”). Отменяет текущие символы командной строки, которая в некоторых случаях при этом стирается. Используется для выхода из текущего режима в прикладных программах. В ряде других ситуаций выводит символ косой черты “\” в конце текущей строки и курсор перебрасывается в начальную позицию следующей строки.

<Shift> – <Верх>, <РЕГ>, <↑> — смена регистра (на время нажатия клавиши). Обычно используются для ввода прописных (заглавных) букв и специальных символов верхнего регистра клавиатуры, для чего нужно нажать клавишу <Shift> и, не отпуская ее, нажать клавишу с обозначением соответствующей буквы или символа.

<Caps Lock> – <Фикс Верх>, <ФПБ>, <ЗГЛ> — нажатие этой клавиши фиксирует клавиатуру в режиме прописных (заглавных) букв, что удобно при вводе заголовков или текста, состоящего из таких букв, так как теперь для их набора не требуется одновременного нажатия клавиши <Shift>. Отметим, что при нажатии клавиши <Shift> в режиме заглавных букв одновременное нажатие любой буквенной клавиши приводит к появлению на экране соответствующей строчной буквы нижнего регистра. При повторном нажатии клавиши <Caps Lock> происходит возврат к режиму строчных букв. Действие клавиши <Caps Lock> охватывает только клавиши с изображением букв. Для ввода же специальных символов верхнего регистра всегда нужно нажимать клавишу <Shift> совместно с соответствующей клавишей.

На компьютерах отечественного производства фиксация регистра русских или латинских букв осуществляется с помощью нажатия клавиш <РУС> или <ЛАТ>, сопровождаемое загоранием соответствующего свето-

вого индикатора. Если на клавиатуре ПК присутствует только одна клавиша <РУС>, то переключение алфавитов осуществляется после каждого ее нажатия. Для одноразового ввода буквы другого алфавита используется клавиша <Р/Л>, которая действует только при нажатии буквенной клавиши. На ряде клавиатур ПК клавиша <Р/Л> также используется для ввода специальных символов, указанных третьими справа по середине клавиш верхнего цифрового ряда.

На компьютерах зарубежного производства это переключение шрифтов определяется видом использованной специальной программы — драйвера клавиатуры, который после нажатия соответствующей комбинации клавиш начинает передавать в компьютер символы другого алфавита. Обычно в качестве переключателя регистра используются клавиши, без которых в принципе можно обойтись. В зависимости от вида используемого драйвера для переключения алфавитов может быть использована клавиша <Caps Lock>, клавиша <F11>, правые клавиши <Ctrl>, <Shift> или одновременное нажатие обеих клавиш <Shift>+<Shift> (правой и левой), одновременное нажатие клавиш <Ctrl>+<Alt>, <Ctrl>+правый <Shift> (в этом случае <Ctrl>+левый <Shift> переключает клавиатуру в режим латинских букв) и т.п.

<CTRL>, <Control> – <УПР> и <ALT> – <ДОП>, <АЛТ> — (“Control” – “Управление” и “Alternate” – “Изменение”) используются только совместно с другими клавишами, изменяя их действие.

<TAB> – <ТАБ>, < > — (табуляция) перемещение курсора по строке на 8 позиций вправо на нижнем регистре и на 8 позиций влево на верхнем регистре (<Shift>+<Tab>).

<Back Space>, <BS>, <Back sp>, <BKSP> – <←> (стрелка влево над клавишей <Enter>), <BШ> — удаление символа слева от курсора с одновременным смещением курсора на одну позицию влево.

<Ins>, <Insert> – <ВСТ> — переключение между двумя режимами ввода символов: вставка или замещение (замена). В режиме вставки курсор принимает форму мигающего квадрата, а не черточки, на месте которого появляется набираемый символ, сдвигающий всю остальную часть строки вправо.

, <Delete> – <УДЛ>, <УД>, <Удал> — (“Вычеркивание”) удаление символа в той позиции, в которой находится курсор (при этом все

символы справа от курсора сдвигаются на одну позицию влево для заполнения освободившегося места).

<HOME> – <НАЧ>, <Начало> — перевод курсора в начало строки.

<END> – <КОН>, <Кнц>, <Конец> — перевод курсора в конец строки.

<PageUp>, <PgUp> – <СтрВв>, <СтрВверх>, <СТР↑> — переход на предыдущую страницу изображаемого на экране текста.

<PageDown>, <PgDn> – <СтрВнз>, <СтрВниз>, <СТР↓> — переход на следующую страницу изображаемого на экране текста.

Клавиши, обозначенные стрелками, используются для перемещения курсора по тексту влево, вверх, вниз и вправо.

<NUM LOCK> – <ЦИФ>, <БлкЦифр> — (“Цифровая блокировка”) — включение/выключение правой малой цифровой клавиатуры (вышеописанной в нецифровом режиме, начиная с клавиши <Ins> – <ВСТ>). В цифровом режиме каждая из клавиш рассматриваемой группы генерирует одну из цифр от 0 до 9-ти или десятичную точку соответственно нанесенному на ней обозначению.

Имена накопителей на дисках (дисководов)

Отметим, что НГМД в MS DOS имеют обычно имена А: и В:, причем А: соответствует первому (главному) накопителю (нижний или левый дисковод). Чтобы облегчить управление огромной памятью НЖМД (винчестера), ее для удобства работы подразделяют на несколько томов или разделов — условных дисков, каждый из которых имеет свою собственную метку наподобие дисковода, которым присваивают имена С:, D: и т.д.. При этом машина не делает различия между разделом накопителя на жестком диске и накопителем на дискетах.

4.2 ДЕЙСТВИЯ ПРИ “ЗАВИСАНИИ” КОМПЬЮТЕРА

Если в процессе работы ПК возникнет ситуация, что компьютер не реагирует на нажатия клавиш или реагирует на них неадекватно, то нужно выйти из состояния “зависания” одним из следующих возрастающих по силе действий:

1. Нажать <Esc> – <КЛЮЧ>.

2. Воспользоваться комбинацией <Ctrl>+<ScrollLock> – <УПР>+<ФСД/СТОП>, следствием чего должно быть прекращение работы выполняемой команды или программы и перевод ПК в режим ожидания ввода команды с клавиатуры. Для набора комбинаций клавиш следует нажать одну из них и, не отпуская ее, нажать другую (или другие). Подобные совместные нажатия клавиш будут обозначаться с помощью знака “+” между ними.

3. Произвести перезагрузку, нажав одновременно <Ctrl>+<Alt>+ – <УПР>+<ДОП>+<УДЛ>.

4. Нажать клавишу <Reset> или <СБРОС> на корпусе системного блока (соответственно на ПК зарубежного или отечественного производства).

5. Выключить компьютер, а затем включить снова. При этом нужно помнить одно общее правило относительно системного блока: выключаться он должен первым перед выключением остальных внешних устройств, а включаться последним после включения всех внешних устройств ПК (т.о. при совмещении дисководов с системным блоком выключение ПК происходит в последовательности: системный блок — монитор, а включение в обратном порядке: монитор — системный блок, а при раздельном конструктивном выполнении дисководов выключение ПК происходит в последовательности: системный блок — дисководы, а включение в обратном порядке: дисководы — системный блок). При включении и выключении ПК следует также убедиться в отсутствии в дисководах дискет, которые могут быть при этом испорчены.

5. ОСНОВЫ ОПЕРАЦИОННОЙ СИСТЕМЫ DOS

Операционная система (ОС) служит для создания среды, в которой происходит диалог пользователя с ЭВМ. Она представляет собой совокупность программ, осуществляющих перевод вводимых команд на язык, понятный для ЭВМ. Существует три разновидности операционной системы DOS, под управлением которой работает IBM-совместимый персональный компьютер (MS DOS, PC DOS и DR DOS). Из-за несущественности различий при первоначальном практическом использовании этих систем они обычно [1] обозначаются одинаково (просто DOS) и описываются одновременно.

5.1 ОСНОВНЫЕ ТЕРМИНЫ И ПОНЯТИЯ

Файл (от слова “File” — “Массив”) — это именованный упорядоченный массив однородной информации, размещенной на внешнем носителе (жестком или гибком диске, магнитной ленте), имеющей определенное функциональное назначение и воспринимаемой операционной системой как единое целое. В виде файлов на диске хранится вся информация. Файлом может быть текст программы на алгоритмическом языке, например, Фортран; набор исходных данных; объектный модуль программы, пригодный для ее понимания ЭВМ и т.п.

Имя файла состоит обычно из собственно имени и расширения, часто называемого типом файла, разделенных между собой точкой. Имя ИФ и тип файла ТФ могут содержать от 1 до 8 и от 0 до 3 символов соответственно, в качестве которых могут использоваться алфавитно-цифровые и некоторые специальные символы.

Наличие расширения в полном имени файла не является обязательным, однако оно обычно используется для описания типа информации, записанной в файле, т.е. для описания типа файла. Обычно файлы, относящиеся к одной основной программе, имеют одинаковые имена, но разные типы. Для некоторых типов информации обычно применяют стандартные или типовые расширения:

- .BAT — командный файл DOS для пакетной обработки, представляющий собой последовательность команд, выполняемых автоматически;
- .COM — команда или программа в машинном коде, пригодные для непосредственного выполнения под управлением DOS (командный файл);
- .DAT — файл данных;
- .EXE — перемещаемая программа, готовая к выполнению под управлением DOS;
- .SYS — системные файлы, являющиеся частью операционной системы и используемые для расширения ее возможностей (например, драйверы), и др.

Каталог (Directory) — это таблица содержимого диска или его части, в которой находится список имен файлов, объединенных по некоторому критерию, с указанием их атрибутов (размера, даты и времени последней зарегистрированной модификации каждого файла). Каталоги файлов создаются для удобства организации файловых систем, так как каждый каталог можно рассматривать как раздел внешней памяти, с содержимым которого можно работать достаточно независимо. При создании каждого файла его имя, тип и другие атрибуты записываются в каком-то каталоге. При этом говорят, что файл находится в данном каталоге. Для создания нового каталога на диске в DOS используется команда *MKDIR* (или *MD*).

Имена каталогов подчиняются тем же требованиям, что и имена файлов, однако возможность расширения имени для каталогов обычно не используется. Имена каталогов указываются в оглавлении большими буквами, тогда как имена файлов маленькими.

Корневой каталог, не имеющий имени, всегда имеется на каждом магнитном диске. В нем регистрируются файлы и каталоги (называемые в этом случае обычно подкаталогами). Корневой каталог обозначается с помощью указателя дисководов, за которым следует обратная наклонная черта “\” (в некоторых версиях DOS вместо знака “\” используется знак “/”). Например, A:\ — корневой каталог гибкого диска на дисковом A, C:\ — корневой каталог жесткого диска (или его раздела — условного диска), расположенного на дисковом C.

Многоуровневые каталоги. В каждом подкаталоге (или каталоге 1-го уровня) корневого каталога, кроме находящихся в нем файлов, также

можно создать произвольное количество других подкаталогов или каталогов 2-го уровня, для которых каталог 1-го уровня будет надкаталогом или родительским каталогом. В каждом созданном подкаталоге (или каталоге 2-го уровня) также можно создать произвольное количество других подкаталогов или каталогов 3-го уровня и т.д. Каждый подкаталог представляет собой обычный файл, содержащий список имен относящихся к нему других файлов, часть которых (или все) также может быть в свою очередь подкаталогами, содержащими соответственно свои файлы и другие подкаталоги.

Цепочки включенных друг в друга каталогов обозначаются их именами, разделяемыми знаком обратной наклонной черты “\”. Если знак “\” стоит перед первым именем каталога, значит выше “по иерархии” находится только корневой каталог данного диска. В этом случае знак “\” как бы отделяет несуществующее имя корневого каталога от имени каталога первого уровня, подчиненного корневому. Примеры цепочек подчиненных каталогов: \P-R — каталог P-R находится на 1-м уровне; \P-R\UPR — каталог UPR находится на 2-м уровне; \P-R\UPR\СТАТИКА — каталог СТАТИКА находится на 3-м уровне.

Так образуется иерархическая древовидная файловая структура, называемая многоуровневой системой каталогов. Слово “иерархическая” ни в каком случае не следует понимать буквально. Ни одному каталогу не присваивается приоритет по отношению к другому. Каждый каталог любого уровня также можно рассматривать как раздел внешней памяти, с содержанием которого можно работать абсолютно независимо, поэтому употребляемые часто обозначения “подкаталог”, “надкаталог” или “родительский каталог” имеют только значение для указания их относительного расположения.

Текущий или рабочий каталог. Каталог (или подкаталог), в котором вы находитесь, называется текущим или рабочим каталогом. В каждый данный момент во внимание принимается только рабочий или текущий каталог, который и становится главным на время работы в нем. Когда вы создаете файл или подкаталог, он помещается в текущий каталог. Можно представить, что каждый диск разделен на столько секций, сколько в нем подкаталогов. Командам доступны только файлы, содержащиеся в текущем подкаталоге, чем достигается возможность независимой работы разных пользователей. В текущем подкаталоге также должны быть

расположены все файлы и подпрограммы, необходимые для работы имеющихся там же программ (или в команде *PATH* должны быть указаны пути доступа к соответствующим каталогам, в которых содержатся необходимые файлы).

Пути и полные имена файлов. Принцип многоуровневой организации памяти на диске приводит к тому, что для указания файла уже недостаточно указать только его имя с соответствующим расширением: одноименные (и притом различные по содержанию) файлы могут находиться в нескольких каталогах. Для точной идентификации файла необходимо указать механизм определения местоположения файла в многоуровневой системе каталогов, в качестве чего указывается путь доступа к файлу. Он представляет собой маршрут по именам каталогов, содержащий последовательный указатель перечня имен каталогов на пути к файлу, разделенных между собой символом “\”.

Путь может начинаться или от корневого каталога, или от текущего. В зависимости от этого маршрут поиска называется соответственно абсолютным или относительным. Если путь доступа начинается с обратной косой черты “\”, то DOS осуществляет поиск файла, начиная с корневого каталога. В противном случае она просматривает указанный путь в поисках файла, начиная от текущего (рабочего) каталога.

Путь доступа к файлу PR2-1.LIS, находящемуся в каталоге 3-го уровня СТАТИКА, из корневого каталога можно указать так:

```
\P-R\UPR\СТАТИКА\pr2-1.lis (5.1)
```

Если файл расположен на жестком диске (дискодов С), а текущим является дискодов А, то полное описание файла, требуемое для доступа к файлу PR2-1.LIS, выглядит следующим образом:

```
C:\P-R\UPR\СТАТИКА\pr2-1.lis (5.2)
```

Путь с включенным в него именем файла полностью идентифицирует расположение файла на диске и образует полное описание файла или идентификатор файла (ИДФ). Отметим, что имя файла также должно быть отделено обратной косой чертой от последнего имени подкаталога. Полное описание файла имеет следующий формат:

```
[диск:] [путь \] имя_файла [.расширение] (5.3)
```

Здесь элементы, заключенные в квадратные скобки, являются необязательными. В этом случае, если в описании не указывается дискодов или

путь, то подразумевается текущий дисковод или текущий каталог. Так в примере (2.1) описатель дисковода не указан, так как текущим является дисковод С и маршрут поиска указывается от корневого каталога, а при работе в текущем каталоге СТАТИКА идентификатором находящегося в нем файла ИДФ будет только имя файла с его расширением (если оно имеется, в нашем примере PR2-1.LIS). Последняя возможность очень удобна. Поэтому всегда в дальнейшем будем подразумевать, что работаем с файлами, находящимися в текущем каталоге.

Подстановочные символы “*” и “?” разрешается использовать в обозначениях имен файлов и их расширений для обозначения сразу нескольких файлов, что называется шаблоном имени файла (“групповым”, “родовым” именем или обозначением имени по маске). Знак “*”, встречаемый в имени или типе файла, обозначает любое число произвольных символов, которые допускаются в именах и типах файлов, количество которых ограничено диапазоном от нуля (т.е. отсутствует вовсе) до значения, дополняющего число указанных в шаблоне символов до их максимально допустимого значения (8 для имени и 3 для типа). Вопросительный знак, встречающийся в имени или типе файла, соответствует любому одному произвольному допустимому символу или его отсутствию. Примеры шаблонов: ABC*.D* — определяет подмножество файлов из существующих на диске в текущем каталоге, имена которых начинаются с ABC и имеют длину от 3-х до 8-ми символов, и расширениями, начинающимися с D и имеющими длину от одного до 3-х символов; *. — все файлы текущего каталога, у которых отсутствует тип; A??B.* — все файлы текущего каталога, начинающиеся с A, заканчивающиеся B и имеющие длину от 2-х до 4-х символов.

Обратим внимание, что DOS *игнорирует все символы в имени и типе файла после подстановочного символа звездочки*, поэтому обозначения *abc.*d и *.* трактуются одинаково, как все файлы текущего каталога.

6. ПРАКТИЧЕСКАЯ РАБОТА НА ПК С ИСПОЛЬЗОВАНИЕМ NORTON COMMANDER

Для удобства практической работы в DOS разработаны специальные операционные программы-оболочки, позволяющие за счет некоторого сокращения широких возможностей DOS существенно увеличить простоту и наглядность выполнения большинства наиболее употребительных ее команд. Вся предыдущая глава нужна в основном для понимания того, что будут делать эти программы-оболочки, а также для оценки достоинств работы с ними. К наиболее популярным сервисным программам относится Norton Commander.

6.1 КРАТКИЕ СВЕДЕНИЯ О NORTON COMMANDER (NC)

Запуск Norton Commander

Запуск осуществляется набором в командной строке NC. Эта команда обычно включается в файл автозапуска AUTOEXEC.BAT, в результате чего после загрузки DOS пользователь сразу оказывается в среде Norton Commander.

Экран при этом оказывается разделенным на два окна, с каждым из которых пользователь может работать независимо. Переход курсора из одного окна в другое осуществляется с помощью клавиши табуляции "ТАВ". Эти окна обычно называют панелями и в них содержатся оглавления каталогов, имена которых (и пути доступа к ним), изображаются сверху каждой панели.

Одна строка в каждом окне содержит необходимую информацию о каком-либо файле или каталоге. При этом первая колонка символов соответствует имени и типу файла, вторая — размеру файла. У подкаталога во второй колонке находится указатель SUB-DIR. Имена файлов в оглавлении каталога выводятся строчными буквами, а подкаталоги для их отличия — заглавными. Обычно сначала выводятся сведения о подкаталогах, а затем — о файлах. В оглавлении подкаталога самую верхнюю строку занимает ссылка на родительский каталог в виде многоточия "..", справа от которых изображается UP-DIR.

Выделенный файл или каталог

Будем называть выделенным такой файл или каталог, который на экране выделен серым цветом (на монохромном дисплее — инверсным изображением).

С помощью клавиш перемещения курсора можно передвигать выделенный участок по панели в пределах текущего каталога, имя которого выведено сверху панели. Нажав клавишу табуляции, можно перевести выделенный участок в другую панель Norton Commander, т.е. сделать текущим находящийся там каталог.

Если нажать клавишу <Alt> и, не отпуская ее, начать набирать первые буквы имени нужного файла, то Norton Commander выделит нужный файл в текущем каталоге, как только будет введено достаточное для его отличия количество букв.

Если выполнить набор имени файла после нажатия клавиш <Alt>+<F7>, то поиск файла будет проведен во всех каталогах текущего диска. В имени файла можно использовать шаблоны “*” и “?”. Если будет найдено несколько файлов, то клавишами вертикального перемещения курсора среди найденных необходимо выделить нужный файл. Затем следует в меню выделить ChDir и нажать <Enter> для перехода в тот каталог, где находится нужный файл.

Если выделить файл с расширением .COM, .EXE или .BAT и нажать <Enter>, то начнется выполнение этого файла.

Выделенная группа файлов

Постоянное выделение файла (т.е. помещение его в группу выделенных) осуществляется нажатием клавиши <Ins>. Для снятия выделения файла следует также нажать <Ins>.

Для выделения группы файлов можно также нажать <+> (плюс на функциональной клавиатуре), называемый для сокращения “серый плюс”, и задать шаблон для выбора имен файлов. В нем можно использовать подстановочные символы “*” и “?”, смысл которых тот же, что и в командах DOS. После нажатия клавиши <Enter> выделится группа файлов, имена которых соответствуют указанному шаблону (при вводе шаблона, установленного по умолчанию *.* , выделится весь подкаталог). Для снятия выделения у этой группы файлов нужно нажать <->, называемый для сокращения “серый минус”, (минус на функциональной клавиатуре) и ввести тот же шаблон.

Работа с дисками и каталогами

Для перехода на другой диск следует нажать <Alt>+<F1> для левой панели или <Alt>+<F2> для правой панели. Затем клавишами горизонтального перемещения курсора следует выбрать из появившегося списка доступных дисков нужный и нажать <Enter>.

Для перехода в содержащийся в оглавлении подкаталог нужно совместить курсор с его именем и нажать клавишу <Enter>. Norton Commander “войдет” в этот каталог и выведет его оглавление.

Для перехода в родительский каталог надо выделить “..” и нажать <Enter> или при любом положении курсора нажать одновременно <Ctrl>+<PageUp>.

Для перехода в другой каталог на том же диске следует нажать комбинацию клавиш <Alt>+<F10>, выделить клавишами перемещения курсора среди появившихся на экране нужный каталог и нажать <Enter>.

Командная строка

Ниже панелей располагается командная строка с приглашением DOS, в которой можно вводить обычные команды или запускать программы. После их выполнения происходит возвращение в среду Norton Commander к ее панелям.

Снятие или возвращение панелей, осуществляемое для просмотра выводимых на экран результатов выполнения команд, достигается совместным нажатием клавиш <Ctrl>+<O>.

Для вывода в командную строку имени выделенного файла на панели следует нажать <Ctrl>+<Enter>, а для очистки командной строки — <Esc>.

Введенные команды запоминаются в памяти в виде последовательного списка. Нажатием клавиш <Ctrl>+<E> или <Ctrl>+<X> в командную строку каждый раз выводится команда, которая была введена соответственно перед или после ранее выполненной команды в списке. Выбранная нужная команда после необходимых изменений может быть отправлена на выполнение.

Для выполнения одной из ранее введенных команд без всяких изменений следует нажать <Alt>+<F8>, выделить с помощью клавиш перемещения курсора в появившемся на экране списке нужную команду и нажать <Enter>.

Назначение функциональных клавиш

- <F1> — выводит подсказку;
- <F2> — вызывает меню пользователя;
- <F3> — просмотр содержимого файла (на нем должен стоять указатель);
- <F4> — редактирование текстового файла (на нем должен стоять указатель);
- <F5> — копирование файла, директории или группы файлов;
- <F6> — переименование или пересылка файлов, директорий или групп файлов;
- <F7> — создание поддиректорий;
- <F8> — удаление файлов, директорий или групп файлов;
- <F9> — выводит в верхней части экрана управляющее меню, позволяющее устанавливать режимы работы и производить некоторые другие действия;
- <F10> — выход из Norton Commander.

6.2 NORTON COMMANDER ПОДРОБНЕЕ

6.2.1 Управление панелями и запуск файлов

Рассмотрим подробнее управление панелями Norton Commander и работу с файлами. На экране обычно отображаются две панели (левая и правая). Вверху расположена строка заголовка, указывающая полный путь к директории, отображаемой в панели, ниже отображается имя диска и подзаголовки NAME. Далее следует список директорий (большие буквы) и файлов (маленькие буквы), причем если он не помещается в одном столбце, то его продолжение выводится в следующем столбце и т.д. У файлов с атрибутами «системный» и (или) «скрытый» между именем и расширением выводится символ « ». Если выводимая директория не корневая, то в начале списка помещается символ «.», позволяющий войти в наддиректорию данной директории. Внизу расположена информационная строка, в которой выводится имя указываемого файла, его размер, дата и время

создания (для директории только имя, дата и время создания), или количество и суммарный объем всех выделенных файлов.

Активная панель, управление указателем

Одна из панелей всегда является активной, у нее заголовок (полный путь) выделен инверсным цветом (светлый фон, темные символы). Если в данной директории есть хотя бы один файл или директория (включая указатель на наддиректорию), то одно из имен также выделено инверсным цветом (иногда говорят, что на этом имени стоит указатель). Указатель можно двигать.

Для этого используются:

- <↓> — перемещает указатель на позицию вниз;
- <↑> — перемещает указатель на позицию вверх;
- <→> — перемещает указатель на столбец вправо;
- <←> — перемещает указатель на столбец влево;
- <Home> — перемещает указатель на первую позицию в списке;
- <End> — перемещает указатель на последнюю позицию в списке;
- <PgUp> — перемещает указатель на экран вверх (используется в тех случаях, когда все файлы не могут из-за своей многочисленности поместиться на экране, так что конец или (и) начало списка оказывается как бы за экраном);
- <PgDn> — перемещает указатель на экран вниз;

Действия при нажатии клавиши <Enter>

Действие Norton Commander при нажатии пользователем клавиши <Enter> зависит от того, имеется что-либо в командной строке внизу экрана или нет. Если командная строка не пуста, то при нажатии клавиши <Enter> будет выполнена команда, содержащаяся в командной строке. Если же командная строка пуста, то действие Norton Commander зависит от того, что выделено на панели:

- если указатель стоит на имени директории, то мы войдем в эту директорию;
- если указатель стоит на символе «..», то мы выйдем из текущей директории в ее наддиректорию;
- если указатель стоит на имени выполняемого файла (с расширением .com или .exe), то произойдет его запуск;

- если указатель стоит на имени пакетного файла (с расширением .bat), то начнется его выполнение;
- если указатель стоит на имени архива, то будет выведено оглавление этого архива;
- если нажать комбинацию клавиш <Ctrl>+<Enter>, то имя, на котором стоит указатель, будет помещено в командную строку, однако никаких действий выполняться не будет. Данная возможность используется в том случае, когда программу необходимо запустить с параметрами. Вы просто добавляете недостающие параметры и нажимаете <Enter>.

Вместо нажатия <Enter> можно дважды щелкнуть мышью. В этом случае вышеуказанные действия будут выполнены независимо от того, содержится что-либо в командной строке внизу экрана или нет.

Приведем сводный список команд управления панелями и работы с файлами.

Управление панелями:

- <Tab> — осуществляет переход между панелями, при этом активной становится другая панель (естественно происходит и смена текущей поддиректории);
- <Ctrl>+<U> — поменять панели местами;
- <Ctrl>+<L> — вывести в неактивной панели сводную информацию о диске и директории активной панели;
- <Ctrl>+<Q> — вывести в неактивной панели содержимое файла, указанного на активной панели;
- <Ctrl>+<Z> — вывести в неактивной панели сводную информацию о директории активной панели.

Работа с файлами:

- <Enter> — осуществляет запуск указанного файла или входение в указанную директорию;
- <Ctrl>+<Enter> — переносит имя выделенного файла (директории) в командную строку;
- <Ins> — выделить/отменить выделение файла;
- Серый <+> — выделить группу файлов;
- Серый <-> — отменить выделение группы файлов;

Для просмотра результатов выполнения программы можно воспользоваться следующими комбинациями клавиш:

- <Ctrl>+<O> — убрать/вывести панели на экран (при первом нажатии панели убираются с экрана, при повторном нажатии появляются обратно);
- <Ctrl>+<P> — убрать/вывести не текущую панель на экран;
- <Ctrl>+<F1> — убрать/вывести левую панель на экран;
- <Ctrl>+<F2> — убрать/вывести правую панель на экран.

Сортировка файлов:

Файлы внутри панели могут располагаться в порядке зависимости от имени, расширения, размера, даты и времени создания, или выводятся в том порядке, в котором они записаны на диск. Переход от одного типа сортировки к другому в активной панели осуществляется с помощью клавиш:

- <Ctrl>+<F3> — сортировать файлы по имени;
- <Ctrl>+<F4> — сортировать файлы по расширению;
- <Ctrl>+<F5> — сортировать файлы по дате и времени создания;
- <Ctrl>+<F6> — сортировать файлы по размеру;
- <Ctrl>+<F7> — не сортировать файлы (выводить в порядке их записи на диск).

Выполнение команд:

- <Alt>+<F10> — вывести дерево каталогов;
- <Esc> — очистить содержимое командной строки или отменить выполнение команды;
- <Alt>+<F8> — просмотр ранее введенных команд DOS;
- <Ctrl>+<E> — просмотр ранее введенных команд DOS в командной строке;
- <Ctrl>+<X> — вывести в командную строку DOS следующую после текущей команду;
- <Ctrl>+<PgUp> — перейти в родительский каталог;
- <Ctrl>+<PgDn> — перейти в подкаталог.

Дисковые операции:

- <Alt>+<F1> — смена текущего диска для левой панели;
- <Alt>+<F2> — смена текущего диска для правой панели;
- <Ctrl>+<R> — обновить содержимое текущей панели.

6.2.2 Редактирование текстового файла

Norton Commander имеет чрезвычайно полезную функцию — возможность редактирования не очень больших текстовых файлов (объемом до 64 кбайт).

Если вы хотите отредактировать уже существующий текстовый файл, то нужно подвести к нему подсветку курсора и нажать клавишу <F4>. Если же вы хотите написать новый текстовый файл, то нажмите <Shift>+<F4>, при этом на экране появится запрос на имя нового файла. Рассмотрим возможности при редактировании.

Ввод текста осуществляется обычным образом. Для перехода на другую строку необходимо нажать <Enter>.

Перемещение курсора:

- <←> (<→>) — на позицию влево (вправо);
- <↑> (<↓>) — на позицию вверх (вниз);
- <Ctrl>+<←> — на слово влево;
- <Ctrl>+<→> — на слово вправо;
- <Ctrl>+<Home> — в начало текста;
- <Ctrl>+<End> — в конец текста;
- <Alt>+<F8> — устанавливает курсор на строку с определенным номером (вам будет предложено его ввести).

Удаление символов:

- <BkSp> () — удалить символ слева (под курсором);
- <Ctrl>+<BkSp> — удалить слово слева (справа);
- <Ctrl>+<T> — удалить слово справа;
- <Ctrl>+<Y> — удалить текущую строку;
- <Ctrl>+<K> — удалить все, начиная от текущей позиции и до конца строки.

Операции с файлами:

- <F2> — записать файл с тем же именем;
- <Shift>+<F2> — записать файл под новым именем (на экране появится соответствующее окно запроса);
- <F9> — распечатать файл на принтере;

- <F10> — выйти из текстового редактора (перед выходом, если файл был изменен, редактор предложит вам записать его);
- <Shift>+<F10> — записать файл с тем же именем и выйти из текстового редактора.

Работа с блоками.

Встроенный редактор Norton Commander позволяет производить операции с блоками текста. Блок — это одна или несколько строк текста, причем строка может входить в блок только целиком.

Для выделения блока нужно установить курсор на первую строку блока и нажать <F3>, затем переместить курсор на последнюю строку блока и еще раз нажать <F3>. Блок при этом будет выводиться инверсным цветом (светлый фон, темные символы).

Команды работы с блоком:

- <F3> — пометить блок;
- <Shift>+<F3> — отменить выделение блока;
- <F5> (<F6>) — копировать (переместить) блок (необходимо установить курсор в то место, куда вы хотите скопировать (переместить) блок, и нажать <F5> (<F6>));
- <F8> — удалить блок;
- <Alt>+<F10> — сохранить блок как файл (при этом на экране появится вопрос, под каким именем вы хотите сохранить блок).

Встроенный редактор позволяет осуществлять поиск строки символов в тексте. При нажатии <F7> появляется окно запроса, в которое вы должны ввести интересующую вас последовательность символов. Если флажок **Case sensitive** сброшен, то регистры при поиске не учитываются (т.е. большие и маленькие буквы не различаются); если же флажок установлен, то поиск идет с учетом регистра. При успешном завершении поиска курсор устанавливается на найденный фрагмент; если же текст не найден, то выдается соответствующее сообщение:

- <F7> — осуществляет поиск строки символов от текущего положения курсора и до конца файла (поиск вперед);
- <Shift>+<F7> — осуществляет поиск строки символов от текущего положения курсора и до начала файла (поиск назад);

<Alt>+<F7> — повторяет поиск с текущими параметрами.

Возможен также поиск и одновременная замена текста. При нажатии <F4> появляется окно запроса, в которое вы должны ввести интересующую вас последовательность символов.

Если флажок **Case sensitive** сброшен, то регистры при поиске не учитываются (т.е. большие и маленькие буквы не различаются); если же флажок установлен, то поиск идет с учетом регистра.

При успешном завершении поиска найденный фрагмент текста выделяется, выдается запрос на замену и варианты ответов:

- **Replace** — заменить данный фрагмент;
- **Skip** — не заменять данный фрагмент;
- **Replace All** — заменить везде без дальнейших запросов;
- **Replace One** — заменить только здесь и прекратить поиск;

Если данная строка не была найдена, то выдается соответствующее сообщение:

<F7> — осуществляет поиск и замену строки символов от текущего положения курсора и до конца текста (поиск вперед);

<Shift>+<F7> — осуществляет поиск и замену строки символов от текущего положения курсора и до начала текста (поиск назад);

<Alt>+<F7> — повторяет поиск и замену с текущими параметрами.

В заключение надо отметить, что встроенный в Norton Commander редактор обладает достаточно ограниченным набором функций, так что его обычно используют для редактирования небольших файлов (длиной не более 50-100 строк).

6.2.3 Назначение функциональных клавиш

При рассмотрении различных возможностей оболочки мы уже касались использования функциональных клавиш. Рассмотрим их сейчас более подробно.

В нижней части экрана выводится подсказка о назначении функциональных клавиш, ее содержимое меняется в зависимости от того, нажата клавиша <Ctrl> или <Alt>. Это вполне естественно, т.к. при этом изменя-

ются функции, вызываемые нажатием клавиш <F1> – <F10>. Клавиши <F11> и <F12> в Norton Commander не используются.

Назначение функциональных клавиш

В самом низу экрана располагается строка, напоминающая о назначении в Norton Commander функциональных клавиш <F1> – <F10>.

<F1> — Help (Помощь). Выдается развитая система подсказок и пояснений о назначении функциональных клавиш при работе с NC.

<F2> — User (User Menu — Меню пользователя). Выводит на экран дисплея функциональное меню, задаваемое пользователем в файле NC.MNU. В нем записан список программ и команд, которые можно вызывать из пользовательского меню.

Структура файла NC.MNU имеет вид (6.1). В его первой строке указывается заглавие с именем программы, в последующих строках — необходимые для этого команды:

```
ИМЯ_ПРОГРАММЫ  
КОМАНДА_1  
...  
КОМАНДА_N
```

(6.1)

Файл NC.MNU является обычным текстовым файлом, поэтому его можно редактировать любым текстовым редактором, что мы и рассмотрим ниже в качестве примера.

Предварительно сделаем несколько замечаний:

- строки с командами должны начинаться двумя пробелами;
- многие программы требуют при запуске, чтобы подкаталог, в котором они расположены, был текущим;
- меню пользователя бывает двух типов:
 - *главное* — файл NC.MNU расположен в подкаталоге NC;
 - *локальное* — файл NC.MNU расположен в текущем подкаталоге.

При нажатии клавиши <F2> вначале ищется локальное меню, а если его нет, то вызывается главное меню.

Отметим, что создать новое и отредактировать уже имеющееся меню можно также с помощью пункта **Menu file edit** (Меню пользователя) подменю **Commands** (Команды) [8, с. 54–56].

Пример. Пусть мы хотим через меню пользователя из любого подкаталога быстро вызывать CAB REDUCE (расположенную у нас на вин-

честере в подкаталоге REDUCE корневого каталога диска С). Для внесения в меню пользователя соответствующей строки, необходимо вызвать на редактирование файл NC.MNU, находящийся в подкаталоге NC, и внести необходимые добавления. Для этого нужно выполнить следующие действия:

- войти в подкаталог NC и совместить подсветку курсора с файлом NC.MNU;
- нажать на клавишу <F4>, вызвав его на редактирование, и внести в него необходимые добавления, записав их после ввода нажатием клавиши <F2> на диск:

REDUCE

```
C:  
CD\REDUCE  
reduce.exe  
CD\
```

(6.2)

Первая строка — заглавие с именем программы, которая будет выводиться на экран при нажатии клавиши <F2> (т.е. при вызове пользовательского меню) в одной из его строк.

Во второй строке записывается досовая команда перехода на диск C:. В третьей — команда CD (change directory — сменить директорию, т.е. подкаталог) для перехода в подкаталог REDUCE. Еще ниже — запуск CAB REDUCE и, наконец, команда CD\ для возвращения в корневую директорию (знак “\” как раз и обозначает “корень”).

Все эти строки с командами должны начинаться двумя пробелами, чтобы NC мог отличить заголовки от команд.

Теперь, выйдя из текстового редактора по <F10>, чтобы из любого подкаталога быстро вызывать CAB REDUCE, достаточно вызвать User Menu клавишей <F2>, совместить подсветку курсора с именем этой программы и нажать клавишу <Enter>.

<F3> — View (Просмотр). Служит для просмотра выделенного с помощью курсора файла. После нажатия клавиши <F3> содержимое файла можно читать с помощью клавиш со стрелками вверх и вниз, <PageUp> и <PageDn>, <HOME> и <END>. Для выхода из режима просмотра следует нажать <F10>.

<F4> — Edit (Редактирование). Предназначена для редактирования выделенного курсором файла. Для создания нового файла необходимо на-

жать клавиши <Shift>+<F4>, в появившемся на экране приглашении набрать имя файла и нажать клавишу <Enter>. Если файл с таким именем не найден, то в появившемся на экране сообщении об этом при выделении курсором варианта **New-file (Новый файл)** следует еще раз нажать <Enter>.

Редактирование файла производится в режиме вставки символов, поэтому клавиша <Ins> не работает. Клавиша <Enter> используется для разбиения строки на две, для чего надо поместить курсор в том месте, где надо разделить строку, и нажать клавишу <Enter>. Если при этом курсор находился в начале или в конце строки, то впереди нее или за ней вставится пустая строка. Для соединения двух строк, надо поместить курсор правее последнего символа первой строки и нажать . Отметим, что при нахождении в позиции курсора или справа от него любого символа, действие клавиши по удалению символа происходит обычным образом. Также описанным ранее образом используются остальные стандартные управляющие клавиши. Клавиша <F2> используется для записи файла на диск с тем же именем. С помощью клавиш <Shift>+<F2> можно записать отредактированный файл под другим именем, которое запрашивается. Клавиша <Esc>, как и <F10>, предназначена для выхода из режима редактирования без записи файла на диск.

Встроенный редактор NC версии 4.0 и выше позволяет *выделять блоки из строк текста*. Для этого надо установить курсор последовательно в первую и последнюю строку блока, каждый раз нажимая при этом <F3>. Затем выделенный блок можно скопировать или переместить в позицию перед курсором, нажав соответственно <F5> или <F6>, а при нажатии <F8> удалить. Одновременное нажатие <Shift>+<F3> отменяет выделение блока текста.

<F5> — **Copy (Копирование)**. Применяется для копирования выделенных файла или группы файлов. В середине экрана NC выдает в окне сообщение, что и куда он намерен копировать. По умолчанию файл или группа файлов копируется в каталог, показанный на другой панели, без изменения имен. Если напечатать в окне только новое имя файла, то копирование будет выполнено в текущем каталоге под указанным именем. При необходимости можно набрать и другое имя каталога, в который надо производить копирование. Затем для копирования надо нажать клавишу <Enter>. Если файл с таким именем в каталоге назначения уже существует, то выдается предупреждение с предложением выбрать один из вариантов: **Overwrite** — скопировать файл на место существующего, **Skip**

— не копировать данный файл, All — копировать все файлы (для группы выделенных) на место существующих файлов без дополнительного предупреждения. Для отмены команды надо нажать клавишу <Esc>.

Если после нажатия клавиши <F5> в появившемся окне в качестве назначения копирования указать логическое устройство *PRN*, то после подключения принтера к ПК и нажатия клавиши <Enter>, будет распечатан выделенный файл.

<F6> — Renmov (Rename/Move — Переименовать/Переместить). Предназначена для переименования или переноса выделенного курсором файла из одного каталога в другой. Для переименования файла или подкаталога после нажатия клавиши <F6> в выведенной строке, куда автоматически перемещается курсор, необходимо набрать его новое имя и нажать <Enter>. Команда переноса файла выполняется так же, как и команда копирования. Отличие состоит в том, что команда переноса стирает исходный файл.

<F7> — Mkdir (Make Directory — Создать каталог). Служит для создания нового подкаталога. Для этого в выводимой на экран строке следует указать имя нового подкаталога. После нажатия клавиши <Enter> в оглавлении текущего каталога появится новый пустой подкаталог с указанным именем.

<F8> — Delete (Уничтожить). Предназначена для стирания файла, выделенного курсором, помеченной группы файлов или пустого подкаталога.

<F9> — Menu. В верхней строке экрана выводится управляющее меню, содержащее режимы работы Norton Commander.

<F10> — Quit (Покинуть). Выход из Norton Commander.

Значение функциональных клавиш при нажатии <Shift>

<Shift>+<F1> — вызывает функцию очистки диска от различного рода временных файлов и .bak копий.

<Shift>+<F2> — вызывает сетевое меню, если, конечно, у вас установлена сеть;

<Shift>+<F3> — просмотр содержимого файла (имя файла будет запрошено);

<Shift>+<F4> — редактирование текстового файла (имя файла будет запрошено);

- <Shift>+<F5> — копирование файла, директории или группы файлов и (или) директорий, имя файла, директории будет запрошено; можно также ввести маску имени файла;
- <Shift>+<F6> — переименование или (и) пересылка файлов, директорий или групп файлов и (или) директорий, имя файла, директории будет запрошено; можно также ввести маску имени файла;
- <Shift>+<F7> — создание поддиректории;
- <Shift>+<F8> — удаление файлов, директорий или групп файлов и (или) директорий, имя файла, директории будет запрошено; можно также ввести маску имени файла;
- <Shift>+<F9> — запись на диск текущих настроек оболочки;
- <Shift>+<F10> — выводит в верхней части экрана управляющее меню, аналогично действию клавиши <F9>.

Значение функциональных клавиш при нажатии <Ctrl>

- <Ctrl>+<F1> — убирает/восстанавливает левую панель;
- <Ctrl>+<F2> — убирает/восстанавливает правую панель;
- <Ctrl>+<F3> — сортировка файлов в текущей панели будет производиться по имени;
- <Ctrl>+<F4> — сортировка файлов в текущей панели будет производиться по расширению;
- <Ctrl>+<F5> — сортировка файлов в текущей панели будет производиться по дате и времени создания;
- <Ctrl>+<F6> — сортировка файлов в текущей панели будет производиться по размеру;
- <Ctrl>+<F7> — сортировка файлов в текущей панели будет производиться не будет (файлы будут выводиться в порядке их записи на диск);
- <Ctrl>+<F8> — сравнение двух директорий и вывод на экран различий между ними;
- <Ctrl>+<F9> — вывод файла на принтер (на файле должен стоять указатель);
- <Ctrl>+<F10> — расщепление длинного файла на несколько файлов меньшего размера или (если выделено несколько файлов) склейка нескольких файлов в один;

Значение функциональных клавиш при нажатии <Alt>

- <Alt>+<F1> — смена текущего диска на левой панели;
- <Alt>+<F2> — смена текущего диска на правой панели;
- <Alt>+<F3> — просмотр содержимого файла (на нем должен стоять указатель), причем имя просмотрщика запрашивается, т.е. просмотр осуществляется с помощью внешней программы;
- <Alt>+<F4> — редактирование текстового файла (на нем должен стоять указатель), причем имя редактора запрашивается, т.е. редактирование осуществляется с помощью внешнего редактора;
- <Alt>+<F5> — создание нового архива;
- <Alt>+<F6> — полная распаковка существующего архива;
- <Alt>+<F7> — поиск файла на диске;
- <Alt>+<F8> — список ранее введенных команд;
- <Alt>+<F9> — переключает монитор из режима 25-ти строк в режим 43-х (для EGA) или 50-ти (для VGA или SVGA);
- <Alt>+<F10> — вызывает на экран дерево директорий для быстрой смены текущей директории.

7. РЕШЕНИЕ ЗАДАЧИ НА ПК И ДОВЕДЕНИЕ ПЕРВОНАЧАЛЬНОГО РЕШЕНИЯ ДО ПРАВИЛЬНОГО

Рассмотрим последовательность действий для создания и редактирования файла программы с последующим его выполнением в CAB REDUCE.

7.1 СЕАНС РАБОТЫ НА ПК С ВЫПОЛНЕНИЕМ ПРОГРАММЫ ДЛЯ CAB REDUCE

Для определенности допустим, что это будет программа 2.1 (2.12), расположенная, например, в файле PR2-1 в подкаталоге \C8 корневого каталога текущего диска, на котором находится и CAB REDUCE.

Конечно, имена подкаталога и файла могут быть *любыми допустимыми именами* (до восьми символов), а имя файла может еще иметь расширение до (трех символов). Эти *ваши имена* нужно подставлять везде ниже соответственно вместо имен подкаталога \C8 и файла PR2-1.

Сначала нужно сделать текущим подкаталог, в котором будет находиться файл программы. Для этого следует подвести подсветку курсора к его имени (\C8) и нажать клавишу <Enter>. Если подкаталог не находится на текущем диске, то предварительно следует нажать <Alt>+<F1> для левой или <Alt>+<F2> для правой панели, клавишами горизонтального перемещения курсора перейти на нужный диск и нажать <Enter>.

Если выделенного для вашей работы подкаталога вообще не существует, то его необходимо создать. Для этого в корневом каталоге текущего диска следует нажать функциональную клавишу <F7> и в выводимой на экран строке следует указать имя нового подкаталога (например, C8). После нажатия клавиши <Enter> в оглавлении текущего каталога появится новый пустой подкаталог с указанным именем C8. Далее действия выполняются согласно приведенного сеанса работы.

1. Для создания или редактирования файла программы в текущем каталоге \C8 необходимо при любом положении курсора нажать клавиши <Shift>+<F4>. Затем в появившемся на экране приглашении набрать имя файла (PR2-1) и нажать клавишу <Enter>. Если файл с таким именем существует, то он вызывается на редактирование, если нет — то в появившемся на экране сообщении об этом выделенным окажется вариант New-file (Новый файл) и нужно еще раз нажать <Enter>.

2. Начать набор текста программы или исправление ошибок. Для удаления символа используются клавиши: *над курсором* — , *влево от курсора* — <BkSp>. Для разделения строки на две (добавление пустой) надо поместить курсор в месте разбиения (в начале или конце строки) и нажать клавишу <Enter>. Для соединения двух строк (удаления пустой), надо поместить курсор правее последнего символа первой строки (над пустой) и нажать .

3. После окончания ввода программы, исправления ошибок и проверки набранной информации необходимо ее сохранить. Для этого нужно:

- нажать <F2> (запись на диск с именем редактируемого файла);
- нажать <Shift>+<F2> и в появившейся строке набрать *нужное имя* и нажать клавишу <Enter> (запись на диск с новым именем файла).

Для выхода из режима редактирования следует нажать клавишу <F10>. На экране опять появляются панели Norton Commander.

7.2 ЗАПУСК ПРОГРАММЫ В CAB REDUCE

Чтобы запустить на выполнение программу в CAB REDUCE, нужно:

- выйти из подкаталога \C8 в корневой каталог;
- подвести подсветку курсора к имени подкаталога \REDUCE и нажать клавишу <Enter>;
- подвести подсветку курсора к системному файлу с именем REDUCE33.EXE или REDUCE33.BAT и нажать клавишу <Enter>.

После появления приглашения в командной строке нужно набрать

1: **in "\c8\pr2-1";**

После этого все результаты будут записаны в подкаталоге \C8 в файл с именем PR2-1.LIS (см. команды 10 и 95 программы 2.1 (2.11)).

Для выхода из REDUCE необходимо набрать в командной строке:

2: **bye;**

На экране опять появляются панели Norton Commander.

Для просмотра результатов следует войти в подкаталог \C8 и после совмещения подсветки курсора с именем файла результатов PR2-1.LIS и нажать клавишу <F3>. Листание файла выполняется с помощью клавиш <PageUp> — вперед к концу файла и <PageDn> — назад к его началу. После окончания просмотра необходимо нажать клавишу <F10>.

ЛИТЕРАТУРА

1. Алексеев А., Евсеев Г., Мураховский В., Симонович С. Новейший самоучитель работы на компьютере. – М.: ДЕСС КОМ, 2000.
2. Антонов С.И., Медведев С.В. Система аналитических вычислений "REDUCE-3". (Основные понятия и конструкции). – М.: МАИ, 1987.
3. Выполнение заданий для курсовых работ по теоретической механике с применением ЭВМ: Метод. пособие по комплекс. преподаванию теор. механики, вычислит. мат. и программирования / [В.М. Носов, Ф.И. Подгайский, А.С. Ковеня и др.; Под общей ред. В.М. Носова]. – Мн.: БГПА, 1992.
4. Гурин Н.И., Скоморохов А.Г. Аналитические вычисления в системе REDUCE: Справ. пособие.– Мн.: Наука и техника, 1989.
5. Денисов В. Windows 95 с самого начала. – СПб.: Питер, 1996.
6. Дьяконов В.П. Справочник по системе символьной математики DERIVE. – М.: "СК Пресс", 1998.
7. Еднерал В.Ф., Крюков А.П., Родионов А.Я. Язык аналитических вычислений REDUCE. – М.: Изд-во Моск. ун-та, 1989.
8. Левин А. Самоучитель работы на компьютере. – М.: Нолидж, 1999.
9. Климов Д.М., Руденко В.М. Методы компьютерной алгебры в задачах механики. – М.: Наука, 1989.
10. Корн Г., Корн Т. Справочник по математике для научных работников и инженеров. – М.: Наука, 1984.
11. Микляев А.П. Настольная книга пользователя IBM PC. – М.: Солон, 1998.
12. Новожилов И.В., Зацепин М.Ф. Типовые расчеты по теоретической механике на базе ЭВМ: Учеб. пособие для вузов. – М.: Высш. шк., 1986.
13. Носов В.М. Выполнение заданий для курсовых работ по теоретической механике с применением ЭВМ: Метод. пособие по комплекс. преподаванию теор. механики, вычислит. мат. и программирования. – Мн.: БПИ – БГПА, 1989 – 1993. – Ч. 1 – 6.

14. Носов В.М. Derive. REDUCE. Практическая работа на ПК (на примерах теор. механики и математики): Учеб. пособие. – Мн.: ТЕХНОПРИНТ, 2001.
15. Носов В.М. Методические указания к выполнению индивидуальных заданий по курсу теор. механики. – Мн.: БПИ, 1984 – 1985. – Ч. 1–4.
16. Носов В.М. Определение скоростей и ускорений с использованием их аналогов для основных (базовых) механизмов: Учеб. –метод. пособие для мех. спец. – Мн.: БГПА, 1994.
17. Носов В.М. Программирование на персональных ЭВМ задач теоретической механики: Учеб. пособие. – Мн.: ТЕХНОПРИНТ, 1997.
18. Позняк Ю.В., Лучко Ю.Ф. Система аналитических вычислений REDUCE. – Мн.: БГУ, 1992.
19. Позняк Ю.В. Аналитические преобразования на ЭВМ в математике и механике: Учеб. пособие. – Мн.: БГУ, 1995.
20. Прудников А.П., Брычков Ю.А., Маричев О.И. Интегралы и ряды. – М.: Наука, 1981.
21. Сборник задач по теор. механике, решаемых с применением ЭВМ. Учеб. пос. для втузов/ [Е.М.Будин и др]. – СПб.: Политехника, 1995.
22. Сборник заданий для курсовых работ по теоретической механике: Учеб. пособие для техн. вузов/ [Яблонский А.А., Норейко С.С., Вольфсон С.А. и др.; Под ред. А.А. Яблонского]. – 3-е изд. – М.: Высш. шк., 1978.
23. Сборник заданий для курсовых работ по теоретической механике: Учеб. пособие для техн. вузов/ [Яблонский А.А., Норейко С.С., Вольфсон С.А. и др.; Под ред. А.А. Яблонского]. – 4-е изд. – М.: Высш. шк., 1985.
24. Теоретическая механика. Вывод и анализ уравнений движения на ЭВМ: Учеб. Пособие для вузов. Ч.1 / В.Г. Веретенников, И.И. Карпов, А.П. Маркеев и др.; Под ред. В.Г. Веретенникова. – М.: Высш. шк., 1990.
25. Hearn A.C. REDUCE USER's Manual. Version 3.0. CA 90406, RAND Co., Santa Monica, April, 1983.
26. Rayna G. REDUCE. Software for Algebraic Computation. London Springer Verlag, 1987.

СОДЕРЖАНИЕ

<i>Предисловие</i>	3
<i>Методические рекомендации</i>	8
<i>Основные обозначения</i>	11
ЧАСТЬ 1 СИСТЕМА АНАЛИТИЧЕСКИХ ВЫЧИСЛЕНИЙ (СAB) REDUCE	12
1. Быстрый старт	12
1.1 Первый сеанс работы	12
1.2 Описание элементов языка	19
1.2.1 Символы и предложения.....	19
1.2.2 Числа	20
1.2.3 Идентификаторы.....	21
1.2.4 Переменные	22
1.3 Имена со скобками	24
1.3.1 Операторы	25
1.3.2 Системные функции	26
1.3.3 Переменные с индексами. Массивы и оператор ARRAY	27
1.3.4 Работа с матрицами. Операторы MATRIX и MAT.....	28
1.4 Команды работы с файлами. Создание, редактирование и использование файлов.	30
1.4.1 Команды OUT и SHUT.....	31
1.4.2 Команда IN	33
2. Решение систем линейных алгебраических уравнений (СЛАУ) в системе REDUCE	38
2.1 Этапы решения СЛАУ и структура программ	39
2.2 Решение задач статики на ПК в САВ REDUCE с “малым” количеством уравнений	42
2.2.1 Формализация уравнений равновесия	42

2.2.2	Решение в виде вещественных и рациональных чисел.....	44
2.2.3	Решение в символьном виде. Исследование влияния вариации нагрузки.....	57
2.2.4	Исследование влияния вариации геометрических факторов...	64
2.3	Решение СЛАУ с “большим” количеством уравнений ..	74
2.3.1	Формализация задачи.....	74
2.3.2	Решение в численном виде	78
2.3.3	Исследование влияния вариации нагрузки.....	83
2.3.4	Исследование влияния вариации геометрических факторов...	87
2.4	Многовариантные задачи.....	93
2.4.1	Формализация уравнений	93
2.4.2	Решение в виде рациональных и вещественных чисел.....	95
2.4.3	Оператор цикла FOR ... DO. Учет особенностей задания C-9	101
2.4.4	Исследование вариации нагрузки	104
2.4.5	Исследование вариации геометрических факторов.....	112
3. ...	<i>Символьное дифференцирование в САВ REDUCE и его использование в задачах кинематики</i>	121
3.1	Оператор дифференцирования DF	121
3.2	.. Постановка задач и структура программ по кинематике точки	123
3.3 Определение скорости и ускорения точки по заданным уравнениям ее движения	125
3.4	... Составление уравнений движения точки и определение ее скорости и ускорения.....	131
3.5 Определение кинематических характеристик при сложном движении точки.....	138
	<i>ЧАСТЬ 2. КРАТКИЕ СВЕДЕНИЯ ДЛЯ ПРАКТИЧЕСКОЙ РАБОТЫ НА ПК</i>	159
4.	<i>Общее описание ПК.....</i>	159

4.1	Сравнительное описание клавиатур зарубежных и отечественных ПК	159
4.2	Действия при “зависании” компьютера.....	162
5.	<i>Основы операционной системы DOS</i>	164
5.1	Основные термины и понятия.....	164
6. ..	<i>Практическая работа на ПК с использованием Norton Commander</i>	169
6.1	Краткие сведения о Norton Commander (NC)	169
6.2	Norton Commander подробнее.....	172
6.2.1	Управление панелями и запуск файлов	172
6.2.2	Редактирование текстового файла	176
6.2.3	Назначение функциональных клавиш	178
7.	<i>Решение задачи на ПК и доведение первоначального решения до правильного</i>	185
7.1	Сеанс работы на ПК с составлением программы для CAB REDUCE.....	185
7.2	Запуск программы в CAB REDUCE.....	186
	<i>ЛИТЕРАТУРА</i>	187

Учебное издание

Носов Валерий Михайлович

Практическое использование САВ REDUCE (на примерах
теоретической механики)

Редактор *В.М. Носов*

Технический редактор *В.М. Носов*

Ответственный за выпуск А.П.Аношко

Подписано в печать 22.11.2000. Формат 60×84¹/₁₆. Бум. офсетная.
Усл. печ. л. 12. Тираж 1000 экз. (1-й з-д: 1–100). Зак. 459.

Издатель: УП «Технопринт». Лицензия ЛВ №380 от 28.04.1999 г.
220027, г. Минск, пр-т Ф. Скорины, 65, корп. 14, оф. 209.
тел./факс 231-86-93, тел. 239-91-57

Налоговая льгота по ОКРБ 007-98, ч.1, 22.11.20.500.

Отпечатано с оригинал-макета на УП «Технопринт».

Лицензия ЛП №203 от 26.01.1998 г.
220027, г. Минск, пр-т Ф. Скорины, 65, корп. 14, оф. 209.
тел./факс 231-86-93, тел. 239-91-57