

к локальной и удаленной памяти, реализуемый посредством технологии NUMA. Оперативная память объемом 16 GB работает с частотой 1GHz. На многоядерной системе установлена операционная система Windows Server 2012 R2 (64 бит). Для реализации КМВП использован механизм User Mode Scheduling. В качестве экспериментальной выбрана задача решения системы линейных алгебраических уравнений (СЛАУ) блочно-параллельными методами Гаусса (методом горизонтальных полос и методом горизонтальных циклических полос), при этом рассмотрены четыре алгоритма – два без использования КМВП и два с использованием КМВП. Эксперименты проводились на СЛАУ размером 15872 строки.

Результаты экспериментов показали, что использование КМВП дает прирост производительности 18% для метода горизонтальных полос при сокращении времени выполнения с 446.7с для 1984 потоков до 375.8с для 3968 потоков, и дает прирост производительности 131.7% для метода горизонтальных циклических полос при сокращении времени выполнения с 881.2с для 992 потоков до 380.3с для 3968 потоков по сравнению с многопоточными реализациями, не использующими КМВП. Это свидетельствует о значительной эффективности кооперативной модели выполнения потоков, которая учитывает особенности решаемой задачи.

УДК 681.3

Многообразие способов представления списков в программах на языке «Пролог»

Ковальков А.Т., Ковалькова И.А.

Белорусский национальный технический университет

Список на «Прологе» – это последовательность элементов, разделенных запятыми и заключенных в квадратные скобки, например, [1, 2, 3, 4, 5]. Список в программах на Прологе может быть представлен многими способами. Некоторые примеры представления списков: **L** – именем, **L**=[1,2,3,4,5]; [**Голова**|**Хвост**] – в виде головы и хвоста, **Голова**=1, **Хвост**=[2,3,4,5]; [] – пустой список; [**X**] – список из одного элемента; [**X1**,**X2**]-список из двух элементов; [**_**,**X2**] – список из двух элементов, значение первого элемента неважно; [**X1**,**X2**|**_**] – в голове списка два элемента и произвольный хвост; [**_**|**T**] – голова списка произвольная и т.д. Выбор рационального способа представления списка при программировании процедуры позволяет сократить код, избавиться от вспомогательных процедур, сделать процедуру более понятной. Рассмотрим хотя бы один пример. Требуется удалить предпоследний

элемент списка. Имея набор элементарных процедур по работе со списками, для решения поставленной задачи пришлось бы выполнить последовательно следующие процедуры: выделить последний элемент с сохранением его в переменной, удалить последний элемент, затем в полученном списке опять удалить последний элемент и добавить в конец полученного списка без двух последних элементов сохраненный последний элемент исходного списка, т.е.

goal

```
ввод_списка(L),
выдел_послед_эл(L,Z),
удал_посл_эл(L,L1),
удал_посл_эл(L1,L2),
добавл_эл_в_конец(Z,L2,Res),
write("Результат=",Res),nl.
```

Однако задачу можно решить сразу одной процедурой, если в процедуре удаления последнего элемента списка

```
удал_посл_эл([_],[]). % удаляется последний элемент
удал_посл_эл([H|T],[Y|T1]):-удал_посл_эл(T,T1).
```

изменить граничное условие (первое предложение процедуры)

```
удал_посл_эл([_,Z],[Z]). % удаляется предпоследний элемент
удал_посл_эл([H|T],[Y|T1]):-удал_посл_эл(T,T1).
```

УДК 681.3.06

Решатель судoku

Ковальков А.Т.

Белорусский национальный технический университет

Судoku – увлекательная логическая игра. Классический судoku представляет собой квадрат 9x9 клеток, который в свою очередь разбит на 9 малых квадратов размером 3x3. Часть клеток судoku изначально заполнено цифрами из диапазона от 1 до 9. Требуется пустые клетки заполнить цифрами от 1 до 9 так, чтобы в каждой горизонтали, вертикали и каждом малом квадрате не было повторений (все цифры использовались по разу). Единого алгоритма решения судoku нет, поэтому были разработаны несколько отдельных алгоритмов, каждый из которых заполняет часть пустых клеток и передает полученное новое состояние матрицы в качестве исходного следующему алгоритму, пока не будут заполнены все клетки, если это достижимо.

Реализованы следующие алгоритмы: проверяются все пустые клетки и если окажется, что в клетке может быть только одно значение из девяти цифр, то она заполняется этой цифрой; перебираются цифры