Национальный технический университет Украины

«Киевский политехнический институт»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

**Лабораторная работа №2**

*по курсу «Компьютерные системы»*

**Изучение работы многопроцессорных ВС с общей памятью**

Выполнили

студенты группы ИВ-73

Захожий Игорь

Шумыло Сергей

Вариант №4

Киев-2010

**Цель работы**

Анализ функциональности и эффективности мультипроцессорных систем (SMP) с основной (разделяемой) памятью.

**Задание**

Вариант №4: LU-разложение матрицы. α = 3, β = 5.

**Краткие теоретические сведения**

LU-разложение — представление матрицы A в виде LU, где L — нижняя треугольная матрица, а U — верхняя треугольная матрица. LU-разложение еще называют LU-факторизацией.

Алгоритм для вычисления LU-разложения приведён ниже.

Будем использовать следующие обозначения для элементов матриц [math](http://images.wikia.com/wikitex/images/b/b2/b22/3ee221b42e124159f074dfdbf38f01.png), [math](http://images.wikia.com/wikitex/images/b/ba/ba7/aac83a138ee9abcc7cbeddd9356f41.png), [math](http://images.wikia.com/wikitex/images/c/c0/c06/18bfcbed650575e37f3a45a7bd0171.png); причем диагональные элементы матрицы [math](http://images.wikia.com/wikitex/images/9/9f/9f6/318b2e375e42c384207e0b31ffcf71.png): [math](http://images.wikia.com/wikitex/images/e/e0/e03/1d81e806ba3358e5386b682a1613e1.png), [math](http://images.wikia.com/wikitex/images/0/06/06c/4a6133f03869e9f22591a42f512811.png). Тогда, если известно LU-разложение матрицы, её определитель можно вычислить по формуле [math](http://images.wikia.com/wikitex/images/1/16/162/d8ff7743467a57106f7982c1e1e411.png)

Найти матрицы [math](http://images.wikia.com/wikitex/images/9/9f/9f6/318b2e375e42c384207e0b31ffcf71.png) и [math](http://images.wikia.com/wikitex/images/e/e9/e94/35e91699ecb58b1f8cb9fbb8990961.png) можно следующим образом(выполнять шаги следует строго по порядку, т.к. следующие элементы находятся с использованием предыдущих):

1. [math](http://images.wikia.com/wikitex/images/f/fd/fd6/ce5f26d50f2024f5b8f500a7406301.png)
2. [math](http://images.wikia.com/wikitex/images/c/c3/c34/245639f5b8448ec100612c74c3a331.png)

Для [math](http://images.wikia.com/wikitex/images/0/0e/0e9/82eae4ac40d47c538f7d25f0211311.png)

1. [math](http://images.wikia.com/wikitex/images/b/b8/b82/34c279b6d878581e825ead03ebd1c1.png)
2. [math](http://images.wikia.com/wikitex/images/6/6b/6bc/16045562f67da72205db842e31a071.png)

В итоге мы получим матрицы — [math](http://images.wikia.com/wikitex/images/9/9f/9f6/318b2e375e42c384207e0b31ffcf71.png) и [math](http://images.wikia.com/wikitex/images/e/e9/e94/35e91699ecb58b1f8cb9fbb8990961.png). В программной реализации данного метода (компактная схема Гаусса) для представления матриц [math](http://images.wikia.com/wikitex/images/9/9f/9f6/318b2e375e42c384207e0b31ffcf71.png) и [math](http://images.wikia.com/wikitex/images/e/e9/e94/35e91699ecb58b1f8cb9fbb8990961.png) можно обойтись всего одним массивом, в котором совмещаются матрицы [math](http://images.wikia.com/wikitex/images/9/9f/9f6/318b2e375e42c384207e0b31ffcf71.png) и [math](http://images.wikia.com/wikitex/images/e/e9/e94/35e91699ecb58b1f8cb9fbb8990961.png). Например вот так(для матрицы размером [math](http://images.wikia.com/wikitex/images/3/3e/3e1/9610818a16129a87ae3e77b5415551.png)):

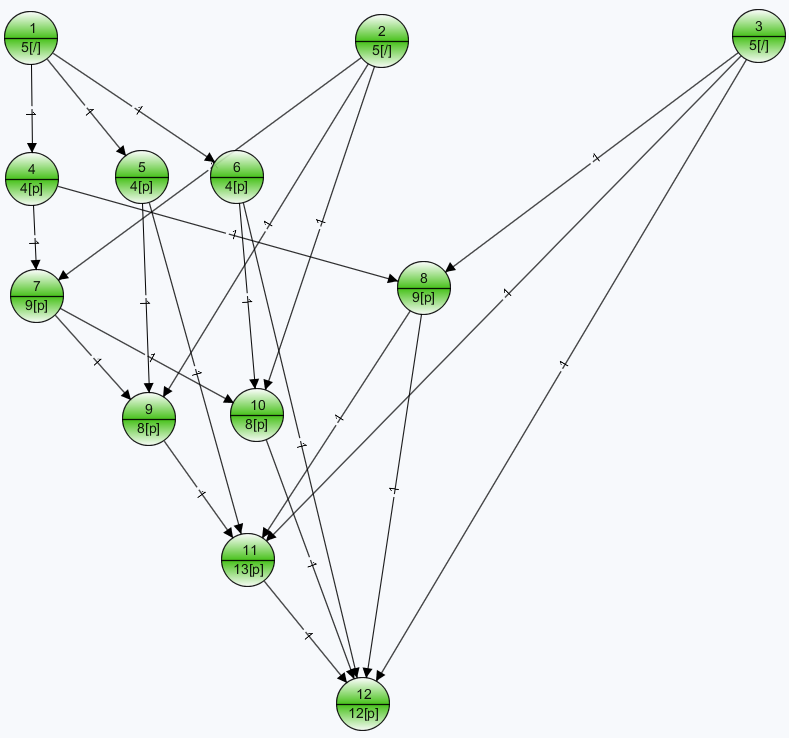
[math](http://images.wikia.com/wikitex/images/9/9e/9eb/78fa0917116f10f7d26c3389937dc1.png)

**Выполнение**

Блок-схема алгоритма:



Ярусно-параллельная форма алгоритма:



**Результаты**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1 банк** | **2 банка** | **3 банка** | **4 банка** |
| **1 процессор** | | | | |
| **t** | 101 | - | - | - |
| **kуск** | 1.0 | - | - | - |
| **kэфф** | 1.0 | - | - | - |
| **2 процессора** | | | | |
| **T** | 68 | 68 | - | - |
| **kуск** | 1.49 | 1.49 | - | - |
| **kэфф** | 0.75 | 0.75 | - | - |
| **3 процессора** | | | | |
| **T** | 65 | 64 | 64 | - |
| **kуск** | 1.55 | 1.58 | 1.58 | - |
| **kэфф** | 0.52 | 0.53 | 0.53 | - |
| **4 процессора** | | | | |
| **t** | 65 | 64 | 64 | 64 |
| **kуск** | 1.55 | 1.58 | 1.58 | 1.58 |
| **kэфф** | 0.52 | 0.53 | 0.53 | 0.53 |

**Выводы**

В данной лабораторной работе мы построили ярусно-параллельную форму алгоритма LU-разложения матрицы и выполнили задачу потактового распределения вычислительных ветвей алгоритма по процессорам ВС. Результаты проверки на программной модели показали, что наибольший коэффициент ускорения, который равен 1.58, мы получаем на SMP-системе, которая имеет 3 процессора и 2 банка памяти. Но коэффициент эффективности при этом равен всего 0.53. Это можно объяснить тем, что в нашем алгоритме только два яруса имеет по три вершины, два – по две и два – по одному. По этой причине часть процессоров иногда простаивает.

В результате выполнения данной лабораторной работы мы можем сделать вывод, что время выполнения задачи, части которой можно распараллелить, можно уменьшить. Этого можно добиться, увеличивая количество процессоров. Увеличение их количества имеет смысл до максимальной степени яруса в алгоритме. При этом будет расти коэффициент ускорения. Дальнейшее увеличение количества процессоров не имеет смысла, так как коэффициент ускорения перестанет расти, а часть процессоров будет простаивать. Однако рост коэффициента ускорения не означает рост коэффициента эффективности, так как некоторые ярусы алгоритма могут иметь степень меньше количества процессоров, и часть процессоров не будут задействованы. Количество банков памяти стоит приблизить к количеству процессоров, чтобы избежать задержек из-за конфликтов доступа к памяти.