

Московский Авиационный Институт
(Национальный Исследовательский Университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

**Лабораторная работа №3 по курсу
«Операционные системы»**

Студент: Путилин Дмитрий
Группа: М80-207Б-21
Вариант: 11
Преподаватель: Миронов Евгений Сергеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2022

Содержание

1. Репозиторий
2. Постановка задачи
3. Общие сведения о программе
4. Общий метод и алгоритм решения
5. Исходный код
6. Демонстрация работы программы
7. Выводы

Репозиторий

<https://github.com/putilin21dn/OC>

Постановка задачи

Цель работы

Целью является приобретение практических навыков в:

- Управление потоками в ОС
- Обеспечение синхронизации между потоками

Задание

Составить программу на языке Си, обрабатывающую данные в многопоточном режиме. При обработке использовать стандартные средства создания потоков операционной системы (Windows/Unix). Ограничение потоков должно быть задано ключом запуска вашей программы.

Так же необходимо уметь продемонстрировать количество потоков, используемое вашей программой с помощью стандартных средств операционной системы.

В отчете привести исследование зависимости ускорения и эффективности алгоритма от входящих данных и количества потоков. Получившиеся результаты необходимо объяснить.

Вариант 11

Наложить K раз фильтры эрозии и наращивания на матрицу, состоящую из вещественных чисел. На выходе получается 2 результирующие матрицы

Общие сведения о программе

Программа компилируется из файла `main.cpp`. Также используются заголовочные файлы: `iostream`, `unistd.h`, `stdio.h`, `stdlib.h`, `vector`, `pthread.h`. В программе используются следующие системные вызовы:

1. `pthread_create()` – создание нового потока
2. `pthread_join()` – ожидание окончания потока

Общий метод и алгоритм решения

На вход подается размер матрицы (используя квадратную) и размер матрицы фильтра. В функции фильтра происходит преобразование. Так каждому потоку передано необходимые строки матрицы. Далее происходит наложения фильтра на матрицу. Так если получается, что фильтр выходит за границы матрицы, эти элементы зануляем и уже с такими проводим действия. Так для элементов из матрицы, попавшие в фильтр, из них находим минимальное число (фильтр эрозии) и максимальное число (фильтр наращивания). После нахождения элемента записываем его в результирующую матрицу на нужное место. Это место, так называемое ядро, определяем просто.

Из выбранных строк мы проходим по каждому элементу и его представляем в виде ядра, а после преобразования на его место ставим найденное число.

Минусы распараллеливания на данной задаче происходит при довольно маленьких размерах матрицы, что при малых потоках вычисления выполняются быстрее нежели на больших количествах потоков.

Исходный код

main.cpp

```
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>
#include <vector>

using namespace std;

struct Data {

    vector<vector<long double> > &mas1;
    vector<vector<long double> > &mas2;
    vector<vector<long double> > &res1;
    vector<vector<long double> > &res2;

    int w_f;
    int thread_num;
    int thread_count;
};

void * multik(void *args){

    vector<vector<long double>>&mas1 = ((Data *)args)->mas1;
    vector<vector<long double>>&mas2 = ((Data *)args)->mas2;
    vector<vector<long double>>&res1 = ((Data *)args)->res1;
    vector<vector<long double>>&res2 = ((Data *)args)->res2;

    int thread_num = ((Data *)args)->thread_num; // номер потока
    int th_count = ((Data *)args)->thread_count; // количество потоков
    int cols = mas1[0].size();
    int offset = ((Data *)args)->w_f/2;
    int rows = mas1.size();

    for (int th_row = thread_num; th_row < rows; th_row += th_count) {

        for (int th_col = 0; th_col < cols; th_col++) {

            long double max = mas1[th_row][th_col];
            long double min = mas2[th_row][th_col];
            for (int i = th_row - offset; i < th_row + offset + 1; i++) {
                for (int j = th_col - offset; j < th_col + offset + 1; j++) {
                    long double curr1, curr2;
                    if ((i < 0) || (i >= rows) || (j < 0) || (j >= cols)) {
                        curr1 = 0;
                    }
                }
            }
        }
    }
}
```

```

        curr2 = 0;
    } else {
        curr1 = mas1[i][j];
        curr2 = mas2[i][j];
    }
    if (curr1 > max) {
        max = curr1;
    }
    if (curr2 < min) {
        min = curr2;
    }
}

res1[th_row][th_col] = max;
res2[th_row][th_col] = min;
}

}

return NULL;
}

int main(int argc, char* argv[]){
    int N, w_f;

    cin >> N;

    vector<vector<long double>> mas1(N, vector<long double>(N,0));
    vector<vector<long double>> mas2(N, vector<long double>(N,0));
    int x;
    for(int i=0; i<N; ++i){
        for(int j=0; j<N; ++j){
            cin >> x;
            mas1[i][j] = x;
            mas2[i][j] = x;
        }
    }

    cin >> w_f;
    int max_thread = atoi(argv[1]);
    vector<vector<long double> > res1(N, vector<long double>(N,0));
    vector<vector<long double> > res2(N, vector<long double>(N,0));

    vector<Data> data = vector<Data>(max_thread, {mas1, mas2, res1, res2, w_f, 0, 0});
    vector<pthread_t> threads = vector<pthread_t>(max_thread);

    for(int k=0; k<atoi(argv[2]); ++k){
        int i=0, j=0;
        for(int p=0; p<max_thread; p++){

            data[p].thread_count = max_thread;
            data[p].thread_num = p;
            if(p>=N*N){
                break;
            }

            if(int err = pthread_create(&threads[p], NULL, multik, (void
*)&data[p])){
                std::cout << "Thread create error: " << err << '\n';
                return -1;
            }
        }
    }
}

```

```

    }

}

//join
for(int i = 0; i < max_thread; i++) {
    if (pthread_join(threads[i],NULL) != 0) {
        cout << "Can't wait for thread\n";
    }
}

//swap
for(int i = 0; i < max_thread; i++) {
    swap(data[i].mas1,data[i].res1);
    swap(data[i].mas2,data[i].res2);
}

cout << "Filtr narashivania \n";
for(int i=0; i<N; ++i){
    for(int j=0;j<N;++j){
        cout << data[0].res1[i][j] << " ";

    }
    cout << '\n';
}

cout << "\nFiltr erosii \n";
for(int i=0; i<N; ++i){
    for(int j=0;j<N;++j){
        cout << data[0].res2[i][j] << " ";
    }
    cout << '\n';
}

}

```

Демонстрация работы программы

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ g++ main.cpp -lpthread

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ cat > test2

5

1 0 1 1 1

1 1 1 0 1

1 1 1 1 1

0 1 0 1 1

1 1 1 0 1

2

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ cat > test

20

4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8

6

6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10
8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12
10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12 14
12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12 14 16
14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12 14 16 18
16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12 14 16 18 20
18 20 22 24 26 28 30 32 0 2 4 6 8 10 12 14 16 18 20 22
20 22 24 26 28 30 32 0 2 4 6 8 10 12 14 16 18 20 22 24
22 24 26 28 30 32 0 2 4 6 8 10 12 14 16 18 20 22 24 26
24 26 28 30 32 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28
26 28 30 32 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30
28 30 32 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32
30 32 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0
32 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6
4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8
6 8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10
8 10 12 14 16 18 20 22 24 26 28 30 32 0 2 4 6 8 10 12

4

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 2 40000 < test2

Filtr narashivania

1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

Filtr erosii

0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

real 0m9,418s

user 0m2,372s

sys 0m12,893s

dmitry@dmitry-VirtualBox:~/Рабочий стол/ОС/lab3/src\$ time ./a.out 12 40000 < test2

Filtr narashivania

1 1 1 1 1

1 1 1 1 1

1 1 1 1 1

1 1 1 1 1

1 1 1 1 1

Filtr erosii

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

real 0m45,375s

user 0m7,716s

sys 1m21,882s

dmitry@dmitry-VirtualBox:~/Рабочий стол/ОС/lab3/src\$ time ./a.out 2 400 < test

Filtr narashivania

12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 12 12 12
14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 14 14 14
16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 16 16 16
18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 16 18 18 18
20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 16 18 20 20 20
22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 22 22
24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 24 24
26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 26 26
28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 28 28
30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 30 30
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 12
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 12 12
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 12 12 12

Filtr erosii

0
0
0 0 4 6 8 10 12 14 16 0 0 0 0 0 0 0 0 0 0 0 0 0


```
006810121416000000000200
0081012141600000000002400
00101214160000000000024600
0012141600000000000246800
0014160000000000024681000
0016000000000002468101200
0000000000000246810121400
00000000000024681012141600
00000000000246810121416000
0000000002468101214160000
0000000024681012141600000
0000000246810121416000000
0000002468101214160000000
0000246810121416000000000
0002468101214160000000000
000000000000000000000000
000000000000000000000000
```

real 0m0,203s

user 0m0,200s

sys 0m0,149s

dmitry@dmitry-VirtualBox:~/Рабочий стол/ОС/lab3/src\$ time ./a.out 12 400 < test

Filtr narashivania

```
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 12 12 12
14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 14 14 14
16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 16 16 16
18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 16 18 18 18
20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 16 18 20 20 20
22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 22 22
24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 24 24
26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 26 26
28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 28 28
30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 30 30
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32 32
32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 16 18 20 22 24 26 28 30 32 32 32 32 32 32
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 12
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 12 12
12 14 16 18 20 22 24 26 28 30 32 32 32 32 32 32 32 32 32 32 32 32 32 12 12 12
```

Filtr erosii

```
000000000000000000000000
000000000000000000000000
00468101214160000000000000
006810121416000000000200
0081012141600000000002400
00101214160000000000024600
0012141600000000000246800
0014160000000000024681000
0016000000000002468101200
0000000000000246810121400
00000000000024681012141600
00000000000246810121416000
0000000002468101214160000
000000024681012141600000
000000246810121416000000
000002468101214160000000
```

```
0 0 0 0 2 4 6 8 10 12 14 16 0 0 0 0 0 0 0 0
0 0 0 2 4 6 8 10 12 14 16 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
real    0m0,357s
user    0m0,263s
sys     0m0,657s
```

```
dmitry@dmitry-VirtualBox:~/Рабочий стол/ОС/lab3/src$ time strace -f -e 'exit' ./a.out 2 10 < test2
```

```
strace: Process 723312 attached
[pid 723312] exit(0)          = ?
strace: Process 723313 attached
[pid 723312] +++ exited with 0 +++
[pid 723313] exit(0)          = ?
[pid 723313] +++ exited with 0 +++
strace: Process 723314 attached
strace: Process 723315 attached
[pid 723314] exit(0)          = ?
[pid 723314] +++ exited with 0 +++
[pid 723315] exit(0)          = ?
[pid 723315] +++ exited with 0 +++
strace: Process 723316 attached
strace: Process 723317 attached
[pid 723316] exit(0)          = ?
[pid 723316] +++ exited with 0 +++
[pid 723317] exit(0)          = ?
[pid 723317] +++ exited with 0 +++
strace: Process 723318 attached
strace: Process 723319 attached
[pid 723318] exit(0)          = ?
[pid 723318] +++ exited with 0 +++
[pid 723319] exit(0)          = ?
[pid 723319] +++ exited with 0 +++
strace: Process 723320 attached
strace: Process 723321 attached
[pid 723320] exit(0)          = ?
[pid 723320] +++ exited with 0 +++
[pid 723321] exit(0)          = ?
[pid 723321] +++ exited with 0 +++
strace: Process 723322 attached
strace: Process 723323 attached
[pid 723322] exit(0)          = ?
[pid 723322] +++ exited with 0 +++
[pid 723323] exit(0)          = ?
[pid 723323] +++ exited with 0 +++
strace: Process 723324 attached
strace: Process 723325 attached
[pid 723324] exit(0)          = ?
[pid 723324] +++ exited with 0 +++
[pid 723325] exit(0)          = ?
[pid 723325] +++ exited with 0 +++
strace: Process 723326 attached
strace: Process 723327 attached
[pid 723326] exit(0)          = ?
[pid 723326] +++ exited with 0 +++
[pid 723327] exit(0)          = ?
[pid 723327] +++ exited with 0 +++
strace: Process 723328 attached
strace: Process 723329 attached
[pid 723328] exit(0)          = ?
```

```
[pid 723328] +++ exited with 0 +++  
[pid 723329] exit(0) = ?  
[pid 723329] +++ exited with 0 +++  
strace: Process 723330 attached  
strace: Process 723331 attached  
[pid 723330] exit(0) = ?  
[pid 723330] +++ exited with 0 +++  
[pid 723331] exit(0) = ?  
[pid 723331] +++ exited with 0 +++
```

Filtr narashivania

```
1 1 1 1 1  
1 1 1 1 1  
1 1 1 1 1  
1 1 1 1 1  
1 1 1 1 1
```

Filtr erosii

```
0 0 0 0 0  
0 0 0 0 0  
0 0 0 0 0  
0 0 0 0 0  
0 0 0 0 0  
+++ exited with 0 +++
```

```
real    0m0,086s  
user    0m0,002s  
sys     0m0,089s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 2 4 < test

```
real    0m0,003s  
user    0m0,000s  
sys     0m0,003s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 2 100 < test

```
real    0m0,026s  
user    0m0,009s  
sys     0m0,019s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 2 10000 < test

```
real    0m1,613s  
user    0m1,639s  
sys     0m1,169s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 12 10000 < test

```
real    0m3,158s  
user    0m2,294s  
sys     0m5,952s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 16 10000 < test

```
real    0m4,111s  
user    0m2,520s  
sys     0m8,062s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 2 100000 < test

```
real    0m15,662s  
user    0m16,218s  
sys     0m11,285s
```

dmitry@dmitry-VirtualBox:~/Рабочий стол/OC/lab3/src\$ time ./a.out 12 100000 < test

11

real 0m33,149s
user 0m23,097s
sys 1m0,175s

Выводы

Составлена и отлажена многопоточная программа на языке Си, выполняющая наложение фильтров эрозии и наращивания на матрицу. Тем самым, приобретены навыки в распараллеливании вычислений, управлении потоками и обеспечении синхронизации между ними.