



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»  
ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування та спеціалізованих комп'ютерних систем**

## **Лабораторна робота №1**

з дисципліни  
**«Паралельні та розподілені обчислення»**

**ТЕМА: «Робота з компіляторами мов С та Java в режимі командного рядка»**

Виконав: студент II курсу  
ФПМ групи КВ-04  
Устименко Ілля Віталійович  
Перевірила:

## Постановка задачі

1. Написати програму розв'язання задачі пошуку (за варіантом) у двовимірному масиві (матриці) одним з алгоритмів методу лінійного пошуку.

Розміри матриці  $m$  та  $n$  взяти самостійно у межах від 7 до 10. Розмір матриці повинен задаватися аргументом запуску програми.

Програма обов'язково повинна бути написана і структурована таким чином:

а) оголошення структур даних (typedef) повинно бути зроблено у окремому заголовочному файлі;

б) повинно бути щонайменше три файли із вихідним кодом (не враховуючи необхідні заголовочні файли), що міститимуть реалізації функцій введення (випадкові значення, наперед сортовані значення, з клавіатури), обробки, та виведення на друк (pretty\_print) елементів матриці;

с) для виконання завдання обробки елементів матриці повинно бути написано дві різні функції:

1) з додатковими операторами виведення налагоджувальної інформації на друк (debug-версія);

2) з виконанням заданих дій без додаткового виведення налагоджувальної інформації (release-версія).

4. Вибір функції повинен робити користувач при запуску програми через аргумент запуску. Наприклад, опція -d вмикає debug режим.

5. Для компіляції написаної багатофайлової програми написати окремий make-файл, причому:

а) при зміні одного із вихідних файлів повинен перекомпільовуватися лише цей файл (а також відбуватися дії, необхідні для генерації бінарного файлу);

б) при видаленні бінарного файлу та незмінних вихідних файлах повинне відбуватися лише лінкування (компоновка бінарного файлу з об'єктних);

с) забезпечити окрему ціль для очистки згенерованих файлів;

6. Забезпечити можливість компіляції написаної багатофайлової програми двома способами:

а) за допомогою однієї команди gcc;

б) за допомогою make-файлу. Марченко О.І., Марченко О.О.

7. Виконати тестування та налагодження програми на комп'ютері. При тестуванні програми необхідно підбирати такі вхідні набори початкових значень матриці, щоб можна було легко відстежити коректність виконання пошуку і ця коректність була б протестована для всіх можливих випадків. З метою тестування дозволяється використовувати матриці меншого розміру.

## Варіант 17

Задано матрицю дійсних чисел  $A[n][n]$ . У побічній діагоналі матриці знайти перший додатний і останній від'ємний елементи, а також поміняти їх місцями.

Програма мовою C  
main.c:

```
#include "process.h"
#include "action.h"
#include "work_with_memory.h"

int main(int argc, char*argv[])
{
    if(work_with_arg(argc,argv) == 'd')
    {
        make_arr();
        debug();
    }
    else
    {
        make_arr();
        realese();
    }
    del_arr();
}
```

---

action.h:

```
#ifndef ACTION_H
#define ACTION_H
void realese();
void debug();
#endif
```

---

action.c:

```
#include"input.h"
#include"pretty_print.h"
#include"process.h"
#include <stdio.h>

void realese()
{
    printf("realese\n");
    printf("-----With random array and relese function-----\n");
    inputRand();
    outputArr();
    findAndSwapRelease();
    outputArr();

    printf("-----With sorted array and release function-----\n");
    inputSorted();
    outputArr();
    findAndSwapRelease();
    outputArr();

    printf("-----With array from keyboard and release function-----\n");
    inputKeyboard();
    findAndSwapRelease();
}
```

```

        outputArr();
    }

    void debug()
    {
        printf("debug\n");
        printf("-----With random array and debug function-----\n");
        inputRand();
        outputArr();
        findAndSwapDebug();
        outputArr();

        printf("-----With sorted array and release function-----\n");
        inputSorted();
        outputArr();
        findAndSwapDebug();
        outputArr();

        printf("-----With array from keyboard and release function-----\n");
        inputKeyboard();
        findAndSwapDebug();
        outputArr();
    }
}

```

---

### input.h:

```

#ifndef INPUT_H
#define INPUT_H
    void inputRand(void);
    void inputSorted(void);
    void inputKeyboard(void);
#endif

```

---

### input.c:

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "typedef.h"

void inputRand()
{
    srand(time(0));
    for(int i = 0; i < how_n; ++i)
    {
        for(int j = 0; j < how_n; ++j)
        {
            arr[i][j] = -40 + rand() % 100;
        }
    }
}

void inputSorted()
{
    int b = 0;
    for(int i = 0; i < how_n; ++i)
    {
        for(int j = 0; j < how_n; ++j)
        {
            arr[i][j] = b;
            ++b;
        }
    }
}

```

```

}

void inputKeyboard()
{
    for(int i = 0; i < how_n; ++i)
    {
        for(int j = 0; j < how_n; ++j)
        {
            scanf("%d",&arr[i][j]);
        }
    }
}

```

---

## process.h:

```

#ifndef PROCESS_H
#define PROCESS_H
void findAndSwapRelease(void);
void findAndSwapDebug(void);
char work_with_arg(int argc, char*argv[]);
#endif

```

---

## process.c:

```

#include <stdlib.h>
#include "typedef.h"
#include <getopt.h>
#include <stdio.h>

char work_with_arg(int argc, char*argv[]) {
    char deb_or_rea='r';
    char take_char;
    while((take_char = getopt(argc,argv,"drn:"))!=-1)
    {
        switch(take_char)
        {
            case 'r':
                deb_or_rea = 'r';
                break;

            case 'd':
                deb_or_rea = 'd';
                break;

            case 'n':
                how_n = atoi(optarg);
                break;
        }
    }
    return deb_or_rea;
}

void findAndSwapRelease() {
    int tmp, firstPositive, ifpos = -1, lastNegative, ilneg = -1;
    for(int i = 0; i < how_n; ++i)
    {
        if(arr[how_n-1-i][i] >= 0)
        {
            firstPositive = arr[how_n-1-i][i];
            ifpos = i;
            break;
        }
    }
    for(int i = 0; i < how_n; ++i) {

```

```

        if(arr[i][how_n-1-i] < 0)
        {
            lastNegative = arr[i][how_n-1-i];
            ilneg = i;
            break;
        }
    }
    if((ifpos != -1)&&(ilneg != -1)){
        tmp = arr[how_n-1-ifpos][ifpos];
        arr[how_n-1-ifpos][ifpos] = arr[ilneg][how_n-1-ilneg];
        arr[ilneg][how_n-1-ilneg] = tmp;
        printf("\nswap\n");
    }else if(ifpos == -
1){printf("\nno_Positive_in_a_mass\n\n");}else{printf("\nno_Negative_in_a_mass\n\n");}
}

void findAndSwapDebug(){
    int tmp,firstPositive,ifpos = -1,lastNegative,ilneg = -1;
    for(int i = 0; i < how_n ; ++i)
    {
        printf("arr[%d][%d]=%d\n",how_n-1-i,i,arr[how_n-1-i][i]);
        if(arr[how_n-1-i][i] >= 0)
        {
            firstPositive = arr[how_n-1-i][i];
            ifpos = i;
            printf("firstPositive = %d\t",firstPositive);
            printf("ifpos=%d",ifpos);
            break;
        }
    }
    printf("\n\n");
    for(int i = 0; i < how_n; ++i) {
        printf("arr[%d][%d]=%d\n",i,how_n-1-i,arr[i][how_n-1-i]);
        if(arr[i][how_n-1-i] < 0)
        {
            lastNegative = arr[i][how_n-1-i];
            ilneg = i;
            printf("lastNegative =%d\t",lastNegative);
            printf("ilneg=%d\n\n",ilneg);
            break;
        }
    }
    if((ifpos != -1)&&(ilneg != -1)) {
        tmp = arr[how_n-1-ifpos][ifpos];
        arr[how_n-1-ifpos][ifpos] = arr[ilneg][how_n-1-ilneg];
        arr[ilneg][how_n-1-ilneg] = tmp;
        printf("sideways diagonal after swapping:");
        for(int i = 0; i < how_n; i++){
            printf("%d\t",arr[how_n-1-i][i]);
        }
    }else if(ifpos == -
1){printf("\nno_Positive_in_a_mass\n\n");}else{printf("\nno_Negative_in_a_mass\n\n");}
    printf("\n\n");
}

```

---

## pretty\_print.h:

```

#ifndef PRETTY_PRINT_H
#define PRETTY_PRINT_H
    void outputArr();
#endif

```

---

## pretty\_print.c:

```
#include <stdio.h>
#include "typedef.h"
void outputArr()
{
    for(int i = 0; i < how_n; ++i)
    {
        for(int j = 0; j < how_n ; ++j)
        {
            printf("%d\t",arr[i][j]);
        }
        printf("\n");
    }
    printf("\n");
}
```

---

## typedef.h:

```
#ifndef TYPEDEF_H
#define TYPEDEF_H
    extern int how_n;
    extern int** arr;
#endif
```

---

## typedef.c:

```
#include "typedef.h"
int** arr;
int how_n = 10;
```

---

## work\_with\_memory.h:

```
#ifndef WORK_WITH_MEMORY_H
#define WORK_WITH_MEMORY_H
void make_arr();
void del_arr();
#endif
```

---

## work\_with\_memory.c:

```
#include<stdlib.h>
#include"typedef.h"

void make_arr()
{
    arr = (int**)malloc(how_n * sizeof(int*));
    for(int i = 0; i < how_n; ++i)
    {
        *(arr+i) = (int*)malloc(how_n * sizeof(int));
    }
}

void del_arr()
{
    for(int i = 0;i < how_n; ++i)
    {
        free(arr[i]);
    }
    free(arr);
}
```

---

## Командні рядки для компілювання та запуску програми мовою C

термінал:

```
gcc -o create main.c action.c input.c pretty_print.c process.c typedef.c  
work_with_memory.c
```

```
./create
```

make build-скомпілювати програму, або якщо якихось файли є-скомпілювати, ті яких нема

make rebuild-перекompілювати повністю програму

make run-скомпілювати і запустити

make debug-скомпілювати в режимі дебаг

make realese-скомпілювати в режимі-реліз

make clean-очистити об'єктні файли і сам бінарний файл

визов дебаг режиму чи розміру матриці

./create -r -n4-приклад

./create -(вид режиму r чи d) -n(розмірності матриці число)

---

makefile:

```
.PHONY: build clean rebuild run greet debug realese  
greet:   
    @echo "Terminating make - please specify target explicitly"  
    @echo "    build    : fast rebuild / build"  
    @echo "    rebuild  : full rebuild"  
    @echo "    run      : run after fast rebuild / build"  
    @echo "    clean    : perform full clean"  
build: create  
rebuild: clean create  
run: build  
    ./create  
  
debug: build  
    ./create -d  
  
realese: build  
    ./create -r  
  
clean:   
    rm -rvf *.o create  
  
main.o: main.c process.h action.h work_with_memory.h  
    gcc -c -o main.o main.c  
process.o: process.c typedef.h  
    gcc -c -o process.o process.c  
action.o: action.c input.h process.h pretty_print.h  
    gcc -c -o action.o action.c  
work_with_memory.o: work_with_memory.c typedef.h  
    gcc -c -o work_with_memory.o work_with_memory.c
```



```
input.o:input.c typedef.h
gcc -c -o input.o input.c
pretty_print.o:pretty_print.c typedef.h
gcc -c -o pretty_print.o pretty_print.c
typedef.o:typedef.c typedef.h
gcc -c -o typedef.o typedef.c
create:main.o process.o action.o work_with_memory.o input.o pretty_print.o typedef.o
gcc -o create main.o process.o action.o work_with_memory.o input.o
pretty_print.o typedef.o
```

---

Тести:

Побудова програми:

```
student@virt-linux:~/kpi/pp/lab_1$ make build
gcc -c -o main.o main.c
gcc -c -o process.o process.c
gcc -c -o action.o action.c
gcc -c -o work_with_memory.o work_with_memory.c
gcc -c -o input.o input.c
gcc -c -o pretty_print.o pretty_print.c
gcc -c -o typedef.o typedef.c
gcc -o create main.o process.o action.o work_with_memory.o input.o pretty_print.
o typedef.o
```

## Тест роботи с параметрами:

```
student@virt-linux:~/kpi/pp/lab_1$ ./create -r -n7
realse
-----With random array and relese function-----
19      39      53      -37      28      -6      29
16      28      56      -35      5       -25     -36
34      32      9       54      40      59      -5
8       2       -37     -37     -24     -35     15
-18     -17     52      41      -37     45      -4
-17     31      -35     39      -1      -39     -4
-4      29      1       22      1       50      17

swap
19      39      53      -37      28      -6      29
16      28      56      -35      5       31      -36
34      32      9       54      40      59      -5
8       2       -37     -37     -24     -35     15
-18     -17     52      41      -37     45      -4
-17     -25     -35     39      -1      -39     -4
-4      29      1       22      1       50      17

-----With sorted array and release function-----
0       1       2       3       4       5       6
7       8       9       10      11      12      13
14      15      16      17      18      19      20
21      22      23      24      25      26      27
28      29      30      31      32      33      34
35      36      37      38      39      40      41
42      43      44      45      46      47      48

no_Negative_in_a_mass
0       1       2       3       4       5       6
7       8       9       10      11      12      13
14      15      16      17      18      19      20
21      22      23      24      25      26      27
28      29      30      31      32      33      34
35      36      37      38      39      40      41
42      43      44      45      46      47      48

-----With array from keyboard and release function-----
_
```

без параметрів:

```
./create
realese
-----With random array and relese function-----
-4      -38      -10      19      -9      53      23      -3      -12      6
4        -34      41       25      59      -2      -40     55      56     -20
-31      6       18       37      -26     -15     -26     11      -6     31
48       30      33      -22     -11     16      23      4       53      3
51       -3      9       -16     14      8       22      14      4     -30
-14      -35     17      -4      42      31      22      57     -18     56
-20      -30     -22      5      40      59     -39     56      55      6
-1       -2      43      1      22      49      1      37      55     45
-1       33      3       56     -31     -15     -21     -17     34      2
31       6      -36     49      3      -3      0       4      -7     48

no_Negative_in_a_mass

-4      -38      -10      19      -9      53      23      -3      -12      6
4        -34      41       25      59      -2      -40     55      56     -20
-31      6       18       37      -26     -15     -26     11      -6     31
48       30      33      -22     -11     16      23      4       53      3
51       -3      9       -16     14      8       22      14      4     -30
-14      -35     17      -4      42      31      22      57     -18     56
-20      -30     -22      5      40      59     -39     56      55      6
-1       -2      43      1      22      49      1      37      55     45
-1       33      3       56     -31     -15     -21     -17     34      2
31       6      -36     49      3      -3      0       4      -7     48

-----With sorted array and relese function-----
0        1        2        3        4        5        6        7        8        9
10       11       12       13       14       15       16       17       18       19
20       21       22       23       24       25       26       27       28       29
30       31       32       33       34       35       36       37       38       39
40       41       42       43       44       45       46       47       48       49
50       51       52       53       54       55       56       57       58       59
60       61       62       63       64       65       66       67       68       69
70       71       72       73       74       75       76       77       78       79
80       81       82       83       84       85       86       87       88       89
90       91       92       93       94       95       96       97       98       99

no_Negative_in_a_mass

0        1        2        3        4        5        6        7        8        9
10       11       12       13       14       15       16       17       18       19
20       21       22       23       24       25       26       27       28       29
30       31       32       33       34       35       36       37       38       39
40       41       42       43       44       45       46       47       48       49
50       51       52       53       54       55       56       57       58       59
60       61       62       63       64       65       66       67       68       69
70       71       72       73       74       75       76       77       78       79
80       81       82       83       84       85       86       87       88       89
90       91       92       93       94       95       96       97       98       99

-----With array from keyboard and relese function-----
```

очищения:

```
student@virt-linux:~/kpi/pp/lab_1$ make clean
rm -rvf *.o create
removed 'action.o'
removed 'input.o'
removed 'main.o'
removed 'pretty_print.o'
removed 'process.o'
removed 'typedef.o'
removed 'work_with_memory.o'
removed 'create'
```

Перебурдова:

```
student@virt-linux:~/kpi/pp/lab_1$ make rebuild
rm -rvf *.o create
removed 'action.o'
removed 'input.o'
removed 'main.o'
removed 'pretty_print.o'
removed 'process.o'
removed 'typedef.o'
removed 'work_with_memory.o'
removed 'create'
gcc -c -o main.o main.c
gcc -c -o process.o process.c
gcc -c -o action.o action.c
gcc -c -o work_with_memory.o work_with_memory.c
gcc -c -o input.o input.c
gcc -c -o pretty_print.o pretty_print.c
gcc -c -o typedef.o typedef.c
gcc -o create main.o process.o action.o work_with_memory.o input.o pretty_print.o typedef.o
student@virt-linux:~/kpi/pp/lab_1$ S
```

Дебаг режим:

```
student@virt-linux:~/kpi/pp/lab_1$ make debug
./create -d
debug
-----With random array and debug function-----
2      38      -32      41      11      -6      46      -15      42      -30
46      -18      16      -24      14      -1      -6      -32      -39      25
20      13      31      6      51      -4      13      15      37      15
-3      -20      -6      5      13      -3      -8      0      -26      -26
10      12      49      18      28      55      58      14      15      11
-21      -24      -36      2      -26      55      38      27      -37      16
35      0      -12      21      -2      41      10      -18      -19      24
48      -16      -23      37      42      -3      -16      -8      52      40
44      -29      8      48      -34      22      -4      44      42      51
52      17      43      -20      30      -19      -38      -20      55      35

arr[9][0]=52
firstPositive = 52      ifpos=0

arr[0][9]=-30
lastNegative = -30      ilneg=0

sideways diagonal after swapping:-30      -29      -23      21      -26      55      -8      15

2      38      -32      41      11      -6      46      -15      42      52
46      -18      16      -24      14      -1      -6      -32      -39      25
20      13      31      6      51      -4      13      15      37      15
-3      -20      -6      5      13      -3      -8      0      -26      -26
10      12      49      18      28      55      58      14      15      11
-21      -24      -36      2      -26      55      38      27      -37      16
35      0      -12      21      -2      41      10      -18      -19      24
48      -16      -23      37      42      -3      -16      -8      52      40
44      -29      8      48      -34      22      -4      44      42      51
-30      17      43      -20      30      -19      -38      -20      55      35

-----With sorted array and release function-----
0      1      2      3      4      5      6      7      8      9
10      11      12      13      14      15      16      17      18      19
20      21      22      23      24      25      26      27      28      29
30      31      32      33      34      35      36      37      38      39
40      41      42      43      44      45      46      47      48      49
50      51      52      53      54      55      56      57      58      59
60      61      62      63      64      65      66      67      68      69
70      71      72      73      74      75      76      77      78      79
80      81      82      83      84      85      86      87      88      89
90      91      92      93      94      95      96      97      98      99

arr[9][0]=90
firstPositive = 90      ifpos=0

arr[0][9]=9
arr[1][8]=18
arr[2][7]=27
arr[3][6]=36
arr[4][5]=45
arr[5][4]=54
arr[6][3]=63
arr[7][2]=72
arr[8][1]=81
arr[9][0]=90

no_Negative_in_a_mass

0      1      2      3      4      5      6      7      8      9
10      11      12      13      14      15      16      17      18      19
20      21      22      23      24      25      26      27      28      29
30      31      32      33      34      35      36      37      38      39
40      41      42      43      44      45      46      47      48      49
50      51      52      53      54      55      56      57      58      59
```

Режим реліз:

```
student@virt-linux:~/kpi/pp/lab_1$ make realese
./create -r
realese
-----With random array and relese function-----
-17    27    -12    -40    38    -26    37    -24    17    -14
32     -11   -16   -35   -35   -21    21   -27    3    36
-36    -20    -6    53    5    15    46   -36   -32   -22
23     43    45    51    44   -24    18    21    -8    27
-1     17     9    24    22   -34    43   -17   -21   -14
12     -17    6    46    28    3     1   -33   -40   -39
-15    23    45    23   -33    29    39    25    42    23
44     -19   -20    -7    45    42    -1   -19    18    18
7      -30    -6     6     8   -38    1     1   -31    1
3      -5     17    40    10    24   -39   -11    41    43

swap
-17    27    -12    -40    38    -26    37    -24    17    3
32     -11   -16   -35   -35   -21    21   -27    3    36
-36    -20    -6    53    5    15    46   -36   -32   -22
23     43    45    51    44   -24    18    21    -8    27
-1     17     9    24    22   -34    43   -17   -21   -14
12     -17    6    46    28    3     1   -33   -40   -39
-15    23    45    23   -33    29    39    25    42    23
44     -19   -20    -7    45    42    -1   -19    18    18
7      -30    -6     6     8   -38    1     1   -31    1
-14    -5     17    40    10    24   -39   -11    41    43

-----With sorted array and relese function-----
0      1      2      3      4      5      6      7      8      9
10     11     12     13     14     15     16     17     18     19
20     21     22     23     24     25     26     27     28     29
30     31     32     33     34     35     36     37     38     39
40     41     42     43     44     45     46     47     48     49
50     51     52     53     54     55     56     57     58     59
60     61     62     63     64     65     66     67     68     69
70     71     72     73     74     75     76     77     78     79
80     81     82     83     84     85     86     87     88     89
90     91     92     93     94     95     96     97     98     99

no_Negative_in_a_mass
0      1      2      3      4      5      6      7      8      9
10     11     12     13     14     15     16     17     18     19
20     21     22     23     24     25     26     27     28     29
30     31     32     33     34     35     36     37     38     39
40     41     42     43     44     45     46     47     48     49
50     51     52     53     54     55     56     57     58     59
60     61     62     63     64     65     66     67     68     69
70     71     72     73     74     75     76     77     78     79
80     81     82     83     84     85     86     87     88     89
90     91     92     93     94     95     96     97     98     99
```

```

-----With random array and relese function-----
7      -33    48     -36    52     46     13     9      -1     46
20     33     42     58     -24    52     -27    52     48     -38
-4     52     48     0      30     47     -31    43     -38    -25
-19    9      -18    -30    13     27     8      18     -24    -1
5      36     -27    39     -13    41     -17    0      33     23
54     22     15     -6     -38    -15    34     24     20     36
39     42     37     14     52     43     -19    0      1      49
-9     -2     -23    4      30     4      37     53     -4     -37
17     -9     25     -28    25     27     50     -1     -9     10
27     23     -8     5      -23    36     -12    50     28     -19

swap
7      -33    48     -36    52     46     13     9      -1     46
20     33     42     58     -24    52     -27    52     48     -38
-4     52     48     0      30     47     -31    43     -38    -25
-19    9      -18    -30    13     27     8      18     -24    -1
5      36     -27    39     -13    41     -17    0      33     23
54     22     15     -6     27     -15    34     24     20     36
39     42     37     14     52     43     -19    0      1      49
-9     -2     -23    4      30     4      37     53     -4     -37
17     -9     25     -28    25     27     50     -1     -9     10
-38    23     -8     5      -23    36     -12    50     28     -19

```