

# Отчет по лабораторной работе №15

по курсу: 1 фундаментальная информатика

студента группы M80-101Б-21 Тулина Ивана, № по списку: 22

Контакты www, e-mail, icq, skype: i.tulin0107@gmail.com

Работа выполнена: «29» декабря 2021г.

Преподаватель: Титов В. К. каф. 806

Входной контроль знаний с оценкой \_\_\_\_\_

Отчет сдан «    » \_\_\_\_\_ 201 \_\_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Работа с матрицами. Динамическое выделение памяти

2. **Цель работы:** Составить программу на языке Си, производящую обработку квадратной матрицы порядка  $N \times N$  ( $1 \leq N \leq 8$ ), из целых чисел, вводимой из стандартного входного текстового файла

3. **Задание (вариант № 22):** перестановка в обратном порядке элементов строки, содержащей минимальный элемент матрицы

4. **Оборудование (лабораторное):**  
ЭВМ \_\_\_\_\_, процессор \_\_\_\_\_, имя узла сети \_\_\_\_\_ с ОП \_\_\_\_\_ Мб,  
НМД \_\_\_\_\_ Мб. Терминал \_\_\_\_\_ адрес \_\_\_\_\_. Принтер \_\_\_\_\_  
Другие устройства \_\_\_\_\_

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Intel Core i5-7300HQ с ОП 7,87 Мб, НМД 15360 Мб. Монитор: встроенный  
Другие устройства \_\_\_\_\_

5. **Программное обеспечение (лабораторное):**  
Операционная система семейства \_\_\_\_\_ наименование \_\_\_\_\_ версия \_\_\_\_\_,  
интерпретатор команд \_\_\_\_\_ версия \_\_\_\_\_  
Система программирования \_\_\_\_\_ версия \_\_\_\_\_  
Редактор текстов \_\_\_\_\_ версия \_\_\_\_\_  
Утилиты операционной системы \_\_\_\_\_

Прикладные системы и программы: \_\_\_\_\_  
Местонахождение и имена файлов программ и данных \_\_\_\_\_

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства UNIX, наименование Ubuntu версия 20.04.3 LTS  
интерпретатор команд bash версия \_\_\_\_\_  
Система программирования \_\_\_\_\_ версия \_\_\_\_\_  
Редактор текстов Emacs версия 3.22.30  
Утилиты операционной системы \_\_\_\_\_  
Прикладные системы и программы \_\_\_\_\_  
Местонахождение и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_

**6. Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

На вход поступает число  $n$  – порядок квадратной матрицы.

I Генерация новой матрицы:

1. Записываем календарное время в переменную `utc`
2. Создаем массив, в который мы будем записывать матрицу
3. В цикле для  $j$  от 0 до  $n$  устанавливаем новую точку начала генерации последовательности псевдослучайных чисел от `utc+j`, генерируем число  $k$  и от него генерируем новую последовательность из  $n$  элементов (элементы очередного столбца матрицы), которые записываем в ранее созданный массив.

II Обработка матрицы

1. Находим минимальный элемент матрицы путем построчного её обхода, в котором происходит сравнение каждого элемента с наименьшим элементом найденным ранее.
2. В строке с минимальным элементом в цикле для  $j$  меняем местами  $j$ -ый элемент и  $n-j-1$  ый элемент

III Вывод обработанной матрицы

**7. Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
int main()
{   int **a,n,k,i,j,c,minel=100,row; long int utc;
    time_t t;
    utc = time(&t);
    printf("Input size matrix : ");
    scanf ("%d", &n);
    a = new int* [n];
    for(i=0; i<n; i++) a[i]=new int [n];

    for(j=0; j<n; j++)
    {srand(utc+j);
     k = rand()%10+15; srand(k);          // генерация матрицы
     for(i=0;i<n;i++) a[i][j]=rand()%10;
    }

    printf ("Started matrix\n");
    for(i=0;i<n;i++)
    { for(j=0; j<n; j++) printf("%3d", a[i][j]);
      printf("\n");
    }

    for(i=0; i<n; i++)
        for(j=0; j<n; j++)
            if(a[i][j]<minel){minel=a[i][j]; row=i;}          // обработка матрицы
    for (j=0; j<n/2; j++){
        c=a[row][j];
        a[row][j]=a[row][n-j-1];
        a[row][n-j-1]=c;}

    printf ("Finish matrix:\n");
    for(i=0; i<n; i++)                                // вывод матрицы
    {   for(j=0; j<n;j++)
        printf(" %3d ", a[i][j]);
        printf("\n");
    }
    return 0;
}
```

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя \_\_\_\_\_

**8. Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

```
yusayu@YS:~/Рабочий стол/cppProjects$ cat head
```

```
*****
*
*      Лабораторная работа №14
*      Работа с матрицами. Динамическое выделение памяти.
*      Выполнил: Тулин Иван Денисович
*      (номер по списку: 22)
*      Группа: М8О-101Б-21
*
*****
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ cat lab15.cpp
```

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
int main()
{   int **a,n,k,i,j,c,minel=100,row; long int utc;
    time_t t;
    utc = time(&t);
    printf("Input size matrix : ");
    scanf ("%d", &n);
    a = new int* [n];
    for(i=0; i<n; i++) a[i]=new int [n];

    for(j=0; j<n; j++)
    {srand(utc+j);
     k = rand()%10+15; srand(k);
     for(i=0;i<n;i++) a[i][j]=rand()%10;
    }

    printf ("Started matrix\n");
    for(i=0;i<n;i++)
    { for(j=0; j<n; j++) printf("%3d", a[i][j]);
      printf("\n");
    }

    for(i=0; i<n; i++)
        for(j=0; j<n; j++)
            if(a[i][j]<minel){minel=a[i][j]; row=i;}
    for (j=0; j<n/2; j++){
        c=a[row][j];
        a[row][j]=a[row][n-j-1];
        a[row][n-j-1]=c;}

    printf ("Finish matrix:\n");
    for(i=0; i<n; i++)
    {   for(j=0; j<n;j++)
        printf(" %3d ", a[i][j]);
        printf("\n");
    }
    return 0;
}
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ g++ -o lab15.out lab15.cpp
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ ./lab15.out
```

```
Input size matrix : 6
```

```
Started matrix
```

```
4 4 9 1 0 3
3 3 6 8 8 9
7 7 8 7 6 6
2 2 2 9 1 9
```

```
9 9 7 6 8 3
3 3 5 1 1 1
```

Finish matrix:

```
3 0 1 9 4 4
3 3 6 8 8 9
7 7 8 7 6 6
2 2 2 9 1 9
9 9 7 6 8 3
3 3 5 1 1 1
```

yusayu@YS:~/Рабочий стол/cppProjects\$ ./lab15.out

Input size matrix : 5

Started matrix

```
9 9 3 2 5
6 6 0 4 7
8 8 7 4 9
2 2 5 4 7
7 7 6 4 9
```

Finish matrix:

```
9 9 3 2 5
7 4 0 6 6
8 8 7 4 9
2 2 5 4 7
7 7 6 4 9
```

yusayu@YS:~/Рабочий стол/cppProjects\$ ./lab15.out

Input size matrix : 1

Started matrix

```
2
```

Finish matrix:

```
2
```

yusayu@YS:~/Рабочий стол/cppProjects\$ ./lab15.out

Input size matrix : 0

Started matrix

Finish matrix:

yusayu@YS:~/Рабочий стол/cppProjects\$

9. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание

10. **Замечания автора** по существу работы \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

11. **Выводы**  
В ходе работы я научился составлять программы на языке Си, использующие динамическое выделение памяти, для обработки матриц.  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Подпись студента \_\_\_\_\_