|  |  |
| --- | --- |
|  | **Отчёт по лабораторной работе** № 15  по курсу: 1 фундаментальная информатика  студент группы : M8O-105Б-21 Титеев Рамиль Маратович , № по списку: 23  Адреса www, e-mail, jabber, skype derol.gym@gmail.com  Работа выполнена: “3 декабря 2021г”  Преподаватель: каф. 806 В. К. Титов  Входной контроль знаний с оценкой  Отчёт сдан “ “ 20 г., итоговая оценка  Подпись преподавателя |

1. **Тема**: Обработка матриц
2. **Цель работы**: Составить программу на языке Си, производящую обработку квадратной матрицы порядка   
   NxN.
3. **Задание (вариант 23)**:

Нахождение суммы элементов строки с минимальным номером, содержащей максимальный элемент матрицы.

1. **Оборудование** (*лабораторное*):

ЭВМ , процессор , имя узла сети с ОП \_ ГБ

НМД \_\_\_ ГБ. Терминал адрес . Принтер

Другие устройства

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Ryzen4600 @ 6x 3.0GHz, ОП 16384 МБ, НМД ГБ. Монитор: встроенный   
Другие устройства

1. **Программное обеспечение** (*лабораторное*):

Операционная система семейства UNIX, наименование: версия \_\_ \_\_

Интерпретатор команд: версия

Система программирования: версия

Редактор текстов: версия

Утилиты операционной системы:

Прикладные системы и программы:

Местонахождения и имена файлов программ и данных:

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства UNIX, наименование Ubuntu версия 20.04

Интерпретатор команд: bash версия

Система программирования: C версия

Редактор текстов: Emacs версия

Утилиты операционной системы:

Прикладные системы и программы:

Местонахождения и имена файлов программ и данных: /usr/bin , a также /bin

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

**В начале мы циклом проходимся по всему массиву и ищем в нем максимальный элемент с минимальным номером строки и запоминаем её индекс. Затем еще одним циклом мы проходимся по строке, индекс которой мы запомнили ранее, и суммируем все элементы в ней.**

**После этого выводим результат.**

1. **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>**

**// Инициализация массива**

**void INIT\_mas(int \*\* &a, int n){**

**a = new int\* [n];**

**for (int i = 0; i < n; i++){**

**a[i] = new int [n];**

**for (int j = 0; j < n; j++){**

**a[i][j] = (rand()%(n\*n) + 1);**

**}**

**}**

**}**

**// Вывод массива**

**void PRINT\_mas(int \*\*a, int n){**

**for (int i = 0; i < n; i++){**

**for (int j = 0; j < n; j++){**

**printf("%d\t", a[i][j]);**

**}**

**printf("\n");**

**}**

**}**

**// Подсчет суммы в массиве по данной строке**

**int SUMM(int \*\*a, int n, int index){**

**int s = 0;**

**for (int i = 0; i<n; i++){**

**s += a[index][i];**

**}**

**return s;**

**}**

**// Функция с решением**

**void SOLVE(int \*\*a, int n){**

**int max\_element = -1, min\_index = n+1;**

**for (int i = 0; i<n; i++){**

**for (int j = 0; j<n; j++){**

**if (a[i][j] > max\_element){**

**max\_element = a[i][j];**

**min\_index = i;**

**}**

**}**

**}**

**printf("\nMax element: %d, min str index: %d", max\_element, min\_index);**

**printf("\nResult: %d\n", SUMM(a, n, min\_index));**

**}**

**int main(){**

**int \*\*a, n;**

**printf("Enter matrix size: ");**

**scanf("%d", &n);**

**INIT\_mas(a, n);**

**printf("\nMassive is created!\n");**

**PRINT\_mas(a, n);**

**SOLVE(a, n);**

**return 0;**

**}**

Допущен к выполнению работы. Подпись преподавателя

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

(base) ramil@ramil:~/labs/lab 15$ cat header.txt

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Лабараторная работа №15 \*

\* Обработка матриц. \*

\* Обход и линеаризация матриц \*

\* Выполнил студент гр. М8О-105-Б \*

\* Титеев Рамиль Маратович \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(base) ramil@ramil:~/labs/lab 15$ cat lab15.cpp

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

// Инициализация массива

void INIT\_mas(int \*\* &a, int n){

a = new int\* [n];

for (int i = 0; i < n; i++){

a[i] = new int [n];

for (int j = 0; j < n; j++){

a[i][j] = (rand()%(n\*n) + 1);

}

}

}

// Вывод массива

void PRINT\_mas(int \*\*a, int n){

for (int i = 0; i < n; i++){

for (int j = 0; j < n; j++){

printf("%d\t", a[i][j]);

}

printf("\n");

}

}

// Подсчет суммы в массиве по данной строке

int SUMM(int \*\*a, int n, int index){

int s = 0;

for (int i = 0; i<n; i++){

s += a[index][i];

}

return s;

}

// Функция с решением

void SOLVE(int \*\*a, int n){

int max\_element = -1, min\_index = n+1;

for (int i = 0; i<n; i++){

for (int j = 0; j<n; j++){

if (a[i][j] > max\_element){

max\_element = a[i][j];

min\_index = i;

}

}

}

printf("\nMax element: %d, min str index: %d", max\_element, min\_index);

printf("\nResult: %d\n", SUMM(a, n, min\_index));

}

int main(){

int \*\*a, n;

printf("Enter matrix size: ");

scanf("%d", &n);

INIT\_mas(a, n);

printf("\nMassive is created!\n");

PRINT\_mas(a, n);

SOLVE(a, n);

return 0;

}

(base) ramil@ramil:~/labs/lab 15$ g++ lab15.cpp

(base) ramil@ramil:~/labs/lab 15$ ./a.out

Enter matrix size: 5

Massive is created!

9 12 3 16 19

11 12 18 25 22

13 3 16 10 14

2 16 2 23 12

12 19 18 5 8

Max element: 25, min str index: 1

Result: 88

(base) ramil@ramil:~/labs/lab 15$ ./a.out

Enter matrix size: 10

Massive is created!

84 87 78 16 94 36 87 93 50 22

63 28 91 60 64 27 41 27 73 37

12 69 68 30 83 31 63 24 68 36

30 3 23 59 70 68 94 57 12 43

30 74 22 20 85 38 99 25 16 71

14 27 92 81 57 74 63 71 97 82

6 26 85 28 37 6 47 30 14 58

25 96 83 46 15 68 35 65 44 51

88 9 77 79 89 85 4 52 55 100

33 61 77 69 40 13 27 87 95 40

Max element: 100, min str index: 8

Result: 638

(base) ramil@ramil:~/labs/lab 15$ ./a.out

Enter matrix size: 3

Massive is created!

2 8 1

8 6 8

2 4 7

Max element: 8, min str index: 0

Result: 11

(base) ramil@ramil:~/labs/lab 15$ ./a.out

Enter matrix size: 6

Massive is created!

20 35 10 8 6 8

11 13 34 2 15 32

15 8 24 23 25 19

17 29 36 9 16 34

27 27 27 32 20 20

6 19 19 31 6 4

Max element: 36, min str index: 3

Result: 141

1. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) 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 |
| 1 | Дом | 28.11.21 | 13:14 | Не работала функция инициализации массива | Прописал адрес массива |  |

1. Замечание автора по существу работы \_\_\_\_\_\_\_\_
2. Выводы Я научился работать с обработкой матриц и выполнять с ними действия, использовать   
   динамическое выделение памяти для матриц.

Подпись студента