МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Практическое занятие № 5\_6.

организация подпрограмм

**Выполнил** студент:

Герасимов Константин Сергеевич

Тюмень-2023

**Работа в аудитории**

**1. Решение задачи 5.1**

***1.1 Постановка задачи***

Вычислить с использованием функции min() элементы каждой строки и столбца матрицы A(m,n). Результаты сформировать в одномерных массивах C(m) и D(n), где m и n не превышают 100. Формат ввода: Двухмерный массив A(m,n)

Формат вывода: Одномерныe массивы C(m) и D(n) содержащие минимальные элементы строк и столбцов соответственно.

***1.2 Текст программы на языке C++***

#include <iostream>  
using namespace std;  
  
void initialize(int\*\* arr, int rows, int columns){  
 srand((time(nullptr)));  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 arr[i][j] = rand() % 80 - 30;  
 }  
 }  
}  
  
void output(int\*\* arr, int rows, int columns){  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 cout.width(3);  
 cout << arr[i][j] << " ";  
 }  
 cout << endl;  
 }  
}  
  
void output\_one\_size(int rows, int arr[rows]){  
 for (int i = 0; i < rows; i++){  
 cout << arr[i] << " ";  
 }  
}  
  
void min(int\*\* arr, int rows, int columns){  
 int min\_row[rows], min\_columns[columns];  
 for (int i = 0; i < rows; i++){  
 int min\_el = arr[i][0];  
 for (int j = 0; j < columns; j++){  
 if (arr[i][j] < min\_el){  
 min\_el = arr[i][j];  
 }  
 }  
 min\_row[i] = min\_el;  
 }  
 for (int i = 0; i < columns; i++){  
 int min\_el = arr[0][i];  
 for (int j = 0; j < rows; j++){  
 if (arr[j][i] < min\_el){  
 min\_el = arr[j][i];  
 }  
 }  
 min\_columns[i] = min\_el;  
 }  
 cout << "Минимальные элементы строк: ";  
 output\_one\_size(rows, min\_row);  
 cout << endl << "Минимальные элементы столбцов: ";  
 output\_one\_size(columns, min\_columns);  
}  
  
int main() {  
 int rows, columns;  
 cout << "Введите колличество строк и столбцов: ";  
 cin >> rows >> columns;  
  
 int\*\* array = new int\* [rows];  
 for (int i = 0; i < rows; i++){  
 array[i] = new int [columns];  
 }  
  
 initialize(array, rows, columns);  
 output(array, rows, columns);  
 cout << endl;  
 min(array, rows, columns);  
}

***1.3 Результат тестирования программы на языке C++***

***Изображение выглядит как текст

Автоматически созданное описание***

**2. Решение задачи 5.3**

***2.1 Постановка задачи***

Определить количество положительных, отрицательных и нулевых элементов матрицы A(i, j) , где i и j непревышают 100. Создать три функции для нахождения этих значений.

Формат ввода: Двухмерный массив A(i ,j)

Формат вывода: Три числа: количество положительных, отрицательных и нулевых элементов матрицы A(i, j)

***2.2 Текст программы на языке C++***

#include <iostream>  
using namespace std;  
  
void initialize(int\*\* arr, int rows, int columns){  
 srand(time(nullptr));  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 arr[i][j] = rand() % 80 - 30;  
 }  
 }  
}  
  
void output(int\*\* arr, int rows, int columns){  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 cout << arr[i][j] << " ";  
 }  
 cout << endl;  
 }  
}  
  
void more\_than\_zero(int\*\* arr, int rows, int columns){  
 int mtz = 0;  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 if (arr[i][j] > 0){  
 mtz++;  
 }  
 }  
 }  
 cout << "Элементов, больших нуль: " << mtz << endl;  
}  
  
void less\_than\_zero(int\*\* arr, int rows, int columns){  
 int ltz = 0;  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 if (arr[i][j] < 0){  
 ltz++;  
 }  
 }  
 }  
 cout << "Элементов, меньших нуль: " << ltz << endl;  
}  
  
void equal\_zero(int\*\* arr, int rows, int columns){  
 int ez = 0;  
 for (int i = 0; i < rows; i++){  
 for (int j = 0; j < columns; j++){  
 if (arr[i][j] == 0){  
 ez++;  
 }  
 }  
 }  
 cout << "Элементов, равных нуль: " << ez << endl;  
}  
  
int main() {  
 int rows, columns;  
 cout << "Введите колличество строк и столбцов: ";  
 cin >> rows >> columns;  
  
 int\*\* array = new int\* [rows];  
 for (int i = 0; i < rows; i++) {  
 array[i] = new int [columns];  
 }  
  
 initialize(array, rows, columns);  
 output(array, rows, columns);  
 more\_than\_zero(array, rows, columns);  
 less\_than\_zero(array, rows, columns);  
 equal\_zero(array, rows, columns);  
}

***2.3 Результат тестирования программы на языке C++***

***Изображение выглядит как текст

Автоматически созданное описание***

**3. Решение задачи 5.5**

***3.1 Постановка задачи***

На шахматной доске стоит конь. Напишите функцию, которая отмечает положение коня на доске и все клетки, которые бьет конь. Клетку, где стоит конь, отметьте буквой “K”, клетки, которые бьет конь, отметьте символами “\*”, остальные клетки заполните точками.

Формат ввода: Программа получает на вход координаты коня на шахматной доске в шахматной нотации (то есть в виде “e4”, где сначала записывается номер столбца (буква от “a” до “h”, слева направо), затем номеру строки (цифра от 1 до 8, снизу вверх).

Формат вывода: Выведите на экран изображение доски.

***3.2 Текст программы на языке C++***

#include <iostream>  
using namespace std;  
  
void initialize(char\*\* arr, char\* cord){  
  
 int coordinats[2] = {(56-int(cord[1])), (49-int(cord[0]))\*(-1)};  
 arr[(56-int(cord[1]))][(49-int(cord[0]))\*(-1)] = 'K';  
  
 for (int i = 0; i < 8; i++){  
 for (int j = 7; j > -1; j--){  
 if ((i == (coordinats[0]-1)) and (j == coordinats[1]-2)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]-1)) and (j == coordinats[1]+2)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]+1)) and (j == coordinats[1]-2)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]+1)) and (j == coordinats[1]+2)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]-2)) and (j == coordinats[1]-1)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]-2)) and (j == coordinats[1]+1)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]+2)) and (j == coordinats[1]-1)) {  
 arr[i][j] = '\*';  
 } else if ((i == (coordinats[0]+2)) and (j == coordinats[1]+1)) {  
 arr[i][j] = '\*';  
 } else if (coordinats[0] == i and coordinats[1] == j){  
 arr[(56-int(cord[1]))][(49-int(cord[0]))\*(-1)] = 'K';  
 } else {  
 arr[i][j] = '.';  
 }  
 }  
 }  
 *//arr[(56-int(cord[1]))][(49-int(cord[0]))\*(-1)] = 'K';* /\*  
 if ((58-int(cord[1])) < 8 or (48-int(cord[0]))\*(-1) < 8){  
 arr[(58-int(cord[1]))][(48-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((54-int(cord[1])) < 8 or (48-int(cord[0]))\*(-1) < 8){  
 arr[(54-int(cord[1]))][(48-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((58-int(cord[1])) < 8 or (50-int(cord[0]))\*(-1) < 8){  
 arr[(58-int(cord[1]))][(50-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((56-int(cord[1])) < 8 or (50-int(cord[0]))\*(-1) < 8){  
 arr[(56-int(cord[1]))][(50-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((57-int(cord[1])) < 8 or (47-int(cord[0]))\*(-1) < 8){  
 arr[(57-int(cord[1]))][(47-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((55-int(cord[1])) < 8 or (47-int(cord[0]))\*(-1) < 8){  
 arr[(55-int(cord[1]))][(47-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((55-int(cord[1])) < 8 or (51-int(cord[0]))\*(-1) < 8){  
 arr[(55-int(cord[1]))][(51-int(cord[0]))\*(-1)] = '\*';  
 }  
 if ((57-int(cord[1])) < 8 or (51-int(cord[0]))\*(-1) < 8){  
 arr[(57-int(cord[1]))][(51-int(cord[0]))\*(-1)] = '\*';  
 }  
  
 PS это мой говно-код :-(  
 \*/  
}  
  
void output(char\*\* arr){  
 for (int i = 0; i < 8; i++){  
 for (int j = 0; j < 8; j++){  
 cout.width(2);  
 cout << arr[i][j] << " ";  
 }  
 cout << endl;  
 }  
}  
  
void horse\_pos(char\* cords){  
 switch (cords[0]) {  
 case 'a':  
 cords[0] = '1';  
 break;  
 case 'b':  
 cords[0] = '2';  
 break;  
 case 'c':  
 cords[0] = '3';  
 break;  
 case 'd':  
 cords[0] = '4';  
 break;  
 case 'e':  
 cords[0] = '5';  
 break;  
 case 'f':  
 cords[0] = '6';  
 break;  
 case 'g':  
 cords[0] = '7';  
 break;  
 case 'h':  
 cords[0] = '8';  
 break;  
 }  
}  
  
int main() {  
 char\*\* desk = new char\* [8];  
 for (int i = 0; i < 8; i++){  
 desk[i] = new char [8];  
 }  
  
 char\* cord = new char [2];  
 cin >> cord;  
 *//cout << int('6')* horse\_pos(cord);  
 initialize(desk, cord);  
 output(desk);  
}

***3.3 Результат тестирования программы на языке C++***

******

**Домашняя работа**

**3. Тестовое задание 1**

***3.1 Задание теста***

Как правильно указать одномерный массив целых чисел в качестве параметра функции?

***3.2 Ответы***

1) void func(int massive){

2) void func(int n, int\* massive[n]){

3) void func(massive){

***3.3 Комментарии к тесту***

***Ответ: 2***

**4. Тестовое задание 2**

***4.1 Задание теста***

Как правильно реализовать вывод целого числа из функции?

***4.2 Ответы***

1) Int func(){

int a;

return a;

}

2) void func(){

int a;

return a;

}

3) int func(int a, int b){

int c = a+b;

}

***4.3 Комментарии к тесту***

Ответ: 1

**5. Тестовое задание 3 (практическая задача)**

***5.1 Формулировка задачи***

Напишите программу, вычисляющую квадраты целых чисел в массиве, от 0 до n-1, где n – размер массива.

***5.2 Ограничения***

Инициализация массива, вывод, функция квадрата – должны быть выполнены в виде отдельных функций.

***5.3 Требования к вводу/выводу***

Ввод: целое число n

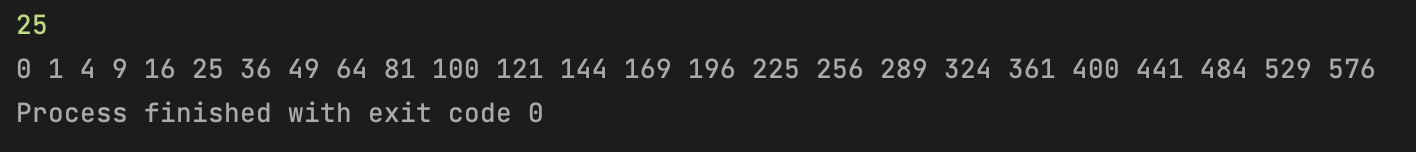
Вывод: массив чисел – квадратов.

***5.4 Оригинальный листинг программы (решение задачи)***

#include <iostream>  
using namespace std;  
  
void squared(int\* arr, int pwr){  
 for (int i = 0; i < pwr; i++){  
 arr[i] = i\*i;  
 }  
}  
  
void output(int\* arr, int pwr){  
 for (int i = 0; i < pwr; i++){  
 cout << arr[i] << " ";  
 }  
}  
  
int main(){  
 int n;  
 cin >> n;  
 int\* massive = new int [n];  
 squared(massive, n);  
 output(massive, n);  
}

***5.5 Набор тестовых данных***

***Изображение выглядит как текст

Автоматически созданное описание***

**6. Список использованных источников**

1. Груздев, Д. В. Программирование C++ (1 курс) : учебное пособие / Д. В. Груздев. — Воронеж : ВГУ, 2017. — 80 с.
2. Основы языка программирования C++. Структура программы. URL: https://metanit.com/cpp/tutorial/2.1.php (Дата обращения. 07.02.2023)