МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 Кафедра Электротехнических комплексов



**Расчётно-графическое задание**

По дисциплине: «Программирование и алгоритмизация»

|  |  |
| --- | --- |
|  | Выполнил:Студент гр. «ЗФ-210», «ИДО»Шифр:140606483Мосеевский Никита Евгеньевич |

Новосибирск

2023

**Содержание**

Задача 1 3

Задача 2 5

Задача 3 8

Задача 4 11

# Задача 1

**Постановка задачи**

3. Составить блок-схему алгоритма и программу вычисления вещественных корней квадратного уравнения *AX*2 + *BX* + *C* = 0.

**Блок-схема программы**

Начало

Ввод a, b, c

d = b \* b - 4 \* a \* c

d < 0

x1 = (-b + sqrt(d)) / (2 \* a)

x2 = (-b - sqrt(d)) / (2 \* a)

Вывод x1, x2

Вывод «Корней нет»

Конец

да

нет

**Текст программы**

#include <iostream>

#include <math.h>

#include <windows.h>

using namespace std;

int main(void)

{

 double a, b, c,//коэффициенты уравнения

 d;//дискриминант квадратного уравнения

 double x1, x2;//корни

 SetConsoleCP(1251);//установка русского шрифта для консоли

 SetConsoleOutputCP(1251);

 cout << "Введите коэффициенты уравнения A, B, C ";//подсказка

 cin >> a >> b >> c; //ввод коэффициентов уравнения

 d = b \* b - 4 \* a \* c;//вычисляем дискриминант

 if (d < 0)//если дискриминант отрицательный

 cout << "Уравнение не имеет вещественных корней." << endl;//корней нет

 else //дискриминант положительный или равный 0, есть корни

 {

 cout << "Корни уравнения:" << endl;

 x1 = (-b + sqrt(d)) / (2.0 \* a);//корни

 x2 = (-b - sqrt(d)) / (2.0 \* a);

 cout << "x1 = " << x1 << endl;//вывод

 cout << "x2 = " << x2 << endl;

 }

 system("pause");//ждем нажатия любой клавиши

 return 0;

}

**Тестовый пример**







# Задача 2

**Постановка задачи**

12. Составить блок-схему алгоритма и программу вычисления количества положительных, отрицательных и нулевых элементов массива *В*[2][3].

**Блок-схема программы**

Начало

Ввод n, m, B

n\_pos = n\_neg = n\_zero = 0

B[i][j] > 0

Конец

да

нет

i = 0, n - 1

j = 0, m - 1

n\_pos = n\_pos + 1

B[i][j] < 0

n\_neg = n\_neg + 1

n\_zero = n\_zero + 1

нет

да

Вывод n\_pos, n\_neg, n\_zero

**Текст программы**

#include <iostream>

#include <windows.h>

using namespace std;

int main(void)

{

 int i, j;//счетчики

 int n = 2, m = 3;//число строк и столбцов массива

 int B[n][m];//двумерный массив

 int n\_pos, n\_neg, n\_zero;//количество положительных, отрицательных и нулевых элементов массива

 SetConsoleCP(1251);//установка русского шрифта для консоли

 SetConsoleOutputCP(1251);

 cout << "Вводите элементы массива" << endl;//подсказка

 for (i = 0; i < n; i++)//ввод массива

 for (j = 0; j < m; j++)

 {

 cout << "B[" << i + 1 << ", " << j + 1 << "] = ";

 cin >> B[i][j];

 }

 n\_pos = n\_neg = n\_zero = 0;//обнуление счетчиков

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

 for (j = 0; j < m; j++)

 if (B[i][j] > 0)//встречен положительный

 n\_pos++;//увеличить счетчик

 else if (B[i][j] < 0)//отрицательный

 n\_neg++;//увеличить счетчик

 else//нулевой

 n\_zero++;//увеличить счетчик

 cout << "Дан массив:" << endl;

 for (i = 0; i < n; i++)//вывод исходного массива

 {

 for (j = 0; j < m; j++)

 cout << B[i][j] << "\t";

 cout << endl;

 }

 cout << "Количество положительных элементов: " << n\_pos << endl;//вывод результата

 cout << "Количество отрицательных элементов: " << n\_neg << endl;

 cout << "Количество нулевых элементов: " << n\_zero << endl;

 system("pause");//ждем нажатия любой клавиши

 return 0;

}

**Тестовый пример**



# Задача 3

**Постановка задачи**

23. Составить блок-схему алгоритма и программу вычисления минимальной компоненты вектора *Y*[5] с использованием функции.

**Блок-схема программы**

Функция вычисления минимальной компоненты вектора

minvector(n, v)

i\_min = 0

v[i] < v[i\_min]

Конец

да

нет

i = 1, n - 1

i\_min = i

minvector = i\_min

Главная программа

Начало

Ввод n, Y

Конец

imin = minvector(n, Y)

Вывод Y[imin], imin

**Текст программы**

#include <iostream>

#include <windows.h>

using namespace std;

int minvector(int n, int \*v)

//вычисление минимальной компоненты вектора

{

 int i\_min;//индекс минимальной компоненты

 int i;//счетчик

 i\_min = 0;//минимум на первом месте

 for (i = 1; i < n; i++)//просмотр массива

 if (v[i] < v[i\_min])//встречен меньший

 i\_min = i;//запомнить номер

 return i\_min;

}

int main(void)

{

 int i;//счетчик

 int n = 5;//размерность массива

 int Y[n];//массив

 int imin;//индекс минимальной компоненты

 SetConsoleCP(1251);//установка русского шрифта для консоли

 SetConsoleOutputCP(1251);

 cout << "Вводите элементы массива" << endl;//подсказка

 for (i = 0; i < n; i++)//ввод массива

 {

 cout << "Y[" << i + 1 << "] = ";

 cin >> Y[i];

 }

 cout << "Дан массив:" << endl;

 for (i = 0; i < n; i++)//вывод исходного массива

 cout << Y[i] << " ";

 cout << endl;

 imin = minvector(n, Y);//вызов функции

 cout << "Минимальный элемент: " << Y[imin] << " в позиции " << imin + 1 << endl;//вывод результата

 system("pause");//ждем нажатия любой клавиши

 return 0;

}

**Тестовый пример**



# Задача 4

**Постановка задачи**

16. Вычисление минимального компонента двумерного массива *Y*[2][3] с использованием функции.

**Блок-схема программы**

Функция вычисления минимальной компоненты двумерного массива

minmatrix(n, m, M, imin, jmin)

imin = jmin = 0

M[i][j] < M[imin][jmin]

Конец

да

нет

i = 0, n - 1

imin = i

jmin = j

j = 0, m - 1

Главная программа

Начало

Ввод n, m, Y

Конец

minmatrix(n, m, Y, imin, jmin)

Вывод Y[imin][jmin], imin, jmin

**Текст программы**

#include <iostream>

#include <windows.h>

using namespace std;

void minmatrix(int n, int m, int \*\*M, int \*imin, int \*jmin)

//вычисление минимальной компоненты двумерного массива

{

 int i, j;//счетчик

 \*imin = 0;//минимум на первом месте

 \*jmin = 0;

 for (i = 0; i < n; i++)//просмотр

 for (j = 0; j < m; j++)

 if (M[i][j] < M[\*imin][\*jmin])//встречен меньший

 {

 \*imin = i;//запомнить номера

 \*jmin = j;

 }

}

int main(void)

{

 int i, j;//счетчики

 int n = 2, m = 3;//число строк и столбцов массива

 int \*\*Y;//двумерный массив

 int imin, jmin;//номер строки и столбца минимального элемента

 SetConsoleCP(1251);//установка русского шрифта для консоли

 SetConsoleOutputCP(1251);

 cout << "Вводите элементы массива" << endl;//подсказка

 Y = new int\*[n];//выделить память под строки

 for (i = 0; i < n; i++)//ввод массива

 {

 Y[i] = new int[m];//выделить память под элементы строки

 for (j = 0; j < m; j++)

 {

 cout << "Y[" << i + 1 << ", " << j + 1 << "] = ";//подсказка

 cin >> Y[i][j];

 }

 }

 cout << "Дан массив:" << endl;

 for (i = 0; i < n; i++)//вывод исходного массива

 {

 for (j = 0; j < m; j++)

 cout << Y[i][j] << "\t";

 cout << endl;

 }

 minmatrix(n, m, Y, &imin, &jmin);//вызов функции

 cout << "Минимальный элемент: " << Y[imin][jmin] << ". Строка: " << imin + 1 << ", столбец: " << jmin + 1 << endl;//вывод результата

 system("pause");//ждем нажатия любой клавиши

 return 0;

}

**Тестовый пример**

