ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Автозаводская высшая школа управления и технологий Кафедра «Прикладная информатика»

Контрольная работа

по дисциплине

«Вычислительная техника и сети в отрасли» Раздел «Программирование С++»

Вариант 20

Выполнил: Новиков Н.А. Студент гр. А21-ТМз(Т) Проверил: Балакина Н.А.

Нижний Новгород 2021

1. **Написать** алгоритмическиформулу

F(a, b) = (arcsin(b + pow(a, 1.4)) / sin(0.5 \* pow(b, a)) + pow(a,7) + exp(0.1 \* a)

1. **Составить** программу (потоковый ввод данных) для вычисления функции

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

wcout << L"Введите x: " << endl;

double x; cin >> x;

wcout << L"Введите y: " << endl;

double y; cin >> y;

double inside\_cos = M\_PI \* x + 4 \* sqrt(y);

double inside\_sin = 2 \* M\_PI - pow(y, 3);

double result = 0.252 \* pow(cos(inside\_cos), 4) + pow(sin(inside\_sin), 2);

wcout << L"Результат вычисления функции Z(x, y) = " << result << endl;

return 1;

}

**Вывод программы:**

Введите x:

0.123

Введите y:

5.13

Результат вычисления функции Z(x, y) = 0.258606

1. **Составить** программу (далее форматный ввод данных) для вычисления таблицы значений функции

при 0.15 a 26 b 8

#include <iostream>

#include <cmath>

#include <iomanip>

using namespace std;

int main()

{

double step\_a = 0.18; double step\_b = 1.;

double left\_a = 0.15; double right\_a = 2.;

double left\_b = 6.; double right\_b = 8.;

cout << "a\tb\tresult" << endl;

/\* If x is a finite value less than 0, and y is a finite noninteger,

a domain error occurs, and a NaN is returned.

Except as specified below, if x or y is a NaN, the result is a NaN.

If x is negative, then large negative or positive y values yield a NaN

as the function result, with errno set to EDOM, and an invalid

(FE\_INVALID) floating-point exception. For example, with pow(),

one sees this behavior when the absolute value of y is greater than about

9.223373e18. \*/

for(double a = left\_a; a <= right\_a; a += step\_a)

{

for(double b = left\_b; b <= right\_b; b += step\_b)

{

double inside\_cos = 1.86 \* b;

double inside\_sin = sqrt(a) \* b;

double result = pow(abs(sin(inside\_sin)), 2.2) - 2. \* pow(cos(inside\_cos), 4.);

cout << setprecision(4) << a << '\t' << b << '\t';

cout << result << endl;

}

}

return 1;

}

**Вывод программы:**

a b result

0.15 6 0.4984

0.15 7 -1.159

0.15 8 -0.4176

0.33 6 0.06953

0.33 7 -0.7421

0.33 8 0.5665

0.51 6 0.8112

0.51 7 -0.3931

0.51 8 -0.1611

0.69 6 0.9197

0.69 7 -1.132

0.69 8 -0.3166

0.87 6 0.3655

0.87 7 -1.261

0.87 8 0.422

1.05 6 0.0107

1.05 7 -0.7318

1.05 8 0.4573

1.23 6 0.1059

1.23 7 -0.3146

1.23 8 -0.1766

1.41 6 0.5228

1.41 7 -0.5184

1.41 8 -0.4153

1.59 6 0.91

1.59 7 -1.023

1.59 8 -0.07497

1.77 6 0.9805

1.77 7 -1.298

1.77 8 0.4513

1.95 6 0.7264

1.95 7 -1.211

1.95 8 0.5478

1. **Составить** программу для вычисления таблицы значений *F(x)* и определить минимальное значение функции. Результат программы записать в файл.

при x=0.912530.6 y 2.5

#include <iostream>

#include <cmath>

#include <fstream>

using namespace std;

int main()

{

double x = 0.91253;

double result = 0.;

double y\_left = 0.6;

double y\_right = 2.4;

double y\_step = 0.8;

ofstream file("..\\result.txt");

if(!file.is\_open()) return 0;

cout << "x\ty\tresult" << endl;

file << "x\ty\tresult" << endl;

for(double y = y\_left; y <= y\_right; y += y\_step)

{

double dividend = sqrt(pow(x,3)) \* y \* y - cos(y);

double divisor = pow(sin(x), 3) - sin(0.2 \* y);

result = dividend / divisor;

cout << x << "\t" << y << "\t" << result << endl;

file << x << "\t" << y << "\t" << result << endl;

}

file.close();

return 1;

}

**Вывод программы:**

x y result

0.91253 0.6 -1.36296

0.91253 1.4 7.03642

0.91253 2.2 69.5986

1. **Составить** программу для определения суммы S положительных элементов массива A[6,8] (элементы массива вводятся с клавиатуры). Заменить значение max элемента массива на значение суммы.

#include <iostream>

#include <vector>

#include <random>

#include <algorithm>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

vector<vector<int>> a;

a.resize(6);

for(int i = 0; i < a.size(); i++)

a[i].resize(8);

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<> distr(-100, 100); // случайные значения в диапазоне 100, 100

wcout << L"Двумерный массив A[6,8] случайных чисел:" << endl << endl;

int sum = 0;

int max\_elem = a[0][0];

int row\_max\_elem, col\_max\_elem;

for(int i = 0; i < a.size(); i++)

{

for(int j = 0; j < a[0].size(); j++)

{

a[i][j] = distr(gen);

cout << a[i][j] << '\t';

if(a[i][j] > 0)

{

sum += a[i][j];

}

if(a[i][j] > max\_elem)

{

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

row\_max\_elem = i;

col\_max\_elem = j;

}

}

cout << endl;

}

wcout << L"\n\nСумма положительных элементов массива A[6,8] S = " << sum << endl;

wcout << L"\nМаксимальный элемент массива = " << max\_elem << endl;

wcout << L"Номер строки максимального элемента = " << row\_max\_elem + 1 << endl;

wcout << L"Номер столбца максимального элемента = " << col\_max\_elem + 1<< endl;

return 1;

}

**Вывод программы:**

Двумерный массив A[6,8] случайных чисел:

12 45 -30 83 -99 19 75 3

92 96 -66 32 -37 6 40 58

-49 -86 15 -34 -33 17 -53 -94

-66 86 94 -97 31 56 -4 -33

-87 -32 -98 -17 69 82 -44 30

84 14 70 -99 12 -43 78 46

Сумма положительных элементов массива A[6,8] S = 1345

Максимальный элемент массива = 96

Номер строки максимального элемента = 2  
Номер столбца максимального элемента = 2

1. **Составить** программу для определения min элемента четных строк массива случайных чисел B[5,5]. Поменять местами min элементы 2 и 3 строки.

#include <iostream>

#include <vector>

#include <random>

#include <algorithm>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

vector<vector<int>> b;

b.resize(5);

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<> distr(1, 100); // случайные значения в диапазоне 1, 100

wcout << L"Двумерный массив B[5,5] случайных чисел:" << endl << endl;

for(int i = 0; i < b.size(); i++)

{

b[i].resize(5);

for(int j = 0; j < b[0].size(); j++)

{

b[i][j] = distr(gen);

cout << b[i][j] << '\t';

}

cout << endl;

}

cout << endl << endl;

vector<int> v;

vector<int> second\_row;

vector<int> third\_row;

for(int i = 0; i < b.size(); i++)

{

for(int j = 0; j < b[0].size(); j++)

{

if(i % 2 != 0)

v.push\_back(b[i][j]);

if(i == 1)

second\_row.push\_back(b[i][j]);

if(i == 2)

third\_row.push\_back(b[i][j]);

}

}

int minimum = \*min\_element(v.begin(), v.end());

wcout << L"Минимальный элемент четных строк массива B[5,5] = " << minimum << endl;

int min\_of\_second\_row = \*min\_element(second\_row.begin(), second\_row.end());

vector<int>::iterator iter\_min\_of\_second\_row = min\_element(second\_row.begin(), second\_row.end());

int pos\_min\_second\_row = distance(second\_row.begin(), iter\_min\_of\_second\_row);

wcout << L"Минимальный элемент второй строки массива B[5,5] = " << min\_of\_second\_row << endl;

int min\_of\_third\_row = \*min\_element(third\_row.begin(), third\_row.end());

vector<int>::iterator iter\_min\_of\_third\_row = min\_element(third\_row.begin(), third\_row.end());

int pos\_min\_third\_row = distance(third\_row.begin(), iter\_min\_of\_third\_row);

wcout << L"Минимальный элемент третьей строки массива B[5,5] = " << min\_of\_third\_row << endl;

int tmp = b[1][pos\_min\_second\_row];

b[1][pos\_min\_second\_row] = b[2][pos\_min\_third\_row];

b[2][pos\_min\_third\_row] = tmp;

wcout << L"\nДвумерный массив B[5,5] случайных чисел после перестановки по условию:" << endl << endl;

for(int i = 0; i < b.size(); i++)

{

for(int j = 0; j < b[0].size(); j++)

cout << b[i][j] << '\t';

cout << endl;

}

return 1;

}

**Вывод программы:**

Двумерный массив B[5,5] случайных чисел:

57 73 35 92 1

60 88 52 96 98

17 67 32 53 71

79 26 8 58 33

34 59 24 4 18

Минимальный элемент четных строк массива B[5,5] = 8

Минимальный элемент второй строки массива B[5,5] = 52

Минимальный элемент третьей строки массива B[5,5] = 17

Двумерный массив B[5,5] случайных чисел после перестановки по условию:

57 73 35 92 1

60 88 17 96 98

52 67 32 53 71

79 26 8 58 33

34 59 24 4 18

1. **Составить** программу для определения min элемента ниже главной диагонали массива случайных чисел C[8,8]. Поменять местами min элемент и

#include <iostream>

#include <cmath>

#include <vector>

#include <random>

#include <algorithm>

#include <tuple>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

vector<vector<int>> m;

m.resize(8);

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_int\_distribution<> distr(1, 100); // случайные значения в диапазоне 1, 100

// Инициализация матрицы случайными числами

vector<tuple<int, int, int>> vt;

wcout << L"Матрица (8 х 8) случайных чисел:" << endl;

for(int i = 0; i < m.size(); i++)

{

m[i].resize(8);

for(int j = 0; j < m.size(); j++)

{

m[i][j] = (int)distr(gen);

cout << m[i][j] << '\t';

if(i > j)

{

tuple<int, int, int> t = make\_tuple(m[i][j], i + 1, j + 1);

vt.push\_back(t);

}

}

cout << endl;

}

cout << endl;

// Сохраняю элементы ниже главной диагонали

vector<int> v;

for(int i = 0; i < vt.size(); i++)

v.push\_back(get<0>(vt[i]));

// Поиск минимального элемента ниже главной диагонали

int min\_elem = \*min\_element(v.begin(), v.end());

vector<int>::iterator it = std::min\_element(v.begin(), v.end());

int iter = distance(v.begin(), it);

int m\_i = get<1>(vt[iter]);

int m\_j = get<2>(vt[iter]);

wcout << L"\nМинимальный элемент ниже главной диагонали = " << min\_elem << endl;

wcout << L"Меняю элемент С(4,2) = " << m[1][3] << L" c элементом С(" << m\_j << ',' << m\_i << ") = " << m[m\_i - 1][m\_j - 1] << endl;

int tmp = m[1][3];

m[1][3] = m[m\_i - 1][m\_j - 1];

m[m\_i - 1][m\_j - 1] = tmp;

wcout << L"Матрица (8 х 8) после смены минимального элемента ниже главной диагонали и С42:" << endl;

for(int i = 0; i < m.size(); i++)

{

for(int j = 0; j < m.size(); j++)

{

cout << m[i][j] << '\t';

}

cout << endl;

}

return 1;

}

**Вывод программы:**

Матрица (8 х 8) случайных чисел:

57 73 35 92 1 60 88 52

96 98 17 67 32 53 71 79

26 8 58 33 34 59 24 4

18 93 97 2 66 78 48 34

7 35 2 42 85 91 29 66

93 58 85 1 57 29 89 73

1 71 27 46 34 23 94 88

48 72 98 89 77 34 71 68

Минимальный элемент ниже главной диагонали = 1

Меняю элемент С(4,2) = 67 c элементом С(4,6) = 1

Матрица (8 х 8) после смены минимального элемента ниже главной диагонали и С42:

57 73 35 92 1 60 88 52

96 98 17 1 32 53 71 79

26 8 58 33 34 59 24 4

18 93 97 2 66 78 48 34

7 35 2 42 85 91 29 66

93 58 85 67 57 29 89 73

1 71 27 46 34 23 94 88

48 72 98 89 77 34 71 68

1. **Составить** программу для создания нового одномерного массива C[5], элементами которого являются минимальные элементы строк массива случайных вещественных чисел D[4,6], найти сумму элементов массива C[5] по формуле   
   Поменять местами первый и последний элементы одномерного массива. Результат записать в файл.

#include <iostream>

#include <vector>

#include <fstream>

#include <random>

#include <algorithm>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

vector<vector<double>> d;

d.resize(4);

std::random\_device rd;

std::mt19937 gen(rd());

std::uniform\_real\_distribution<> distr(1., 100.); // случайные значения в диапазоне 1, 100

wcout << L"Двумерный массив D[4,6] случайных вещественных чисел:" << endl << endl;

for(int i = 0; i < d.size(); i++)

{

d[i].resize(6);

for(int j = 0; j < d[0].size(); j++)

{

d[i][j] = distr(gen);

cout << d[i][j] << '\t';

}

cout << endl;

}

vector<double> v;

for(int i = 0; i < d.size(); i++)

for(int j = 0; j < d[0].size(); j++)

v.push\_back(d[i][j]);

wcout << L"\nОтсортированные элементы двумерного массива D[4,6]:" << endl << endl;

sort(v.begin(), v.end());

for(int i = 0; i < v.size(); i++)

cout << v[i] << '\t';

wcout << L"\n\nМассив C[5] минимальных элементов из D[4,6]:" << endl << endl;

vector<double> c;

double sum = 0;

for(int i = 0; i < 5; i++)

{

c.push\_back(v[i]);

cout << c[i] << '\t';

sum += pow(sin(c[i]), 2);

}

wcout << L"\n\nСумма элементов массива C[5] по формуле в условии = " << sum;

double tmp = c.front();

c.front() = c.back();

c.back() = tmp;

ofstream file("..\\result.txt");

if(file.is\_open())

{

wcout << L"\n\nМассив C[5] минимальных элементов из D[4,6] после перестановки первого и последнего элемента:" << endl << endl;

file << "Массив C[5] минимальных элементов из D[4,6] после перестановки первого и последнего элемента:" << endl << endl;

for(int i = 0; i < 5; i++)

{

cout << c[i] << '\t';

file << c[i] << '\t';

}

}

file.close();

return 1;

}

**Вывод программы:**

Двумерный массив D[4,6] случайных вещественных чисел:

72.8987 91.6186 59.8381 52.0205 97.5398 66.4945

53.3366 79.0608 8.33597 33.6552 58.7906 4.06624

92.9308 2.95549 78.1578 34.2684 34.9101 42.0671

90.824 65.4965 57.4924 1.77499 29.4083 73.2271

Отсортированные элементы двумерного массива D[4,6]:

1.77499 2.95549 4.06624 8.33597 29.4083 33.6552 34.2684 34.9101 42.0671 52.0205 53.3366 57.4924 58.7906 59.8381 65.4965

66.4945 72.8987 73.2271 78.1578 79.0608 90.824 91.6186 92.9308 97.5398

Массив C[5] минимальных элементов из D[4,6]:

1.77499 2.95549 4.06624 8.33597 29.4083

Сумма элементов массива C[5] по формуле в условии = 3.23672

Массив C[5] минимальных элементов из D[4,6] после перестановки первого и последнего элемента:

29.4083 2.95549 4.06624 8.33597 1.77499