

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 C++»

Вариант 20

Выполнил: Новиков Н.А.

Студент гр. А21-ТМз(Т)

Проверил: Балакина Н.А.

Нижний Новгород

2021

1. **Написать** алгоритмически формулу

$$F(a, b) = \frac{\arcsin(b + a^{\frac{1}{4}})}{\sin(0.5b^a)} + a^7 * e^{0.1a}$$

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

2. **Составить** программу (поточный ввод данных) для вычисления функции

$$Z(x, y) = 0.252 \cos^4 \left(\frac{\pi}{2} * x + 4\sqrt{y} \right) + \sin^2(2\pi - y^3)$$

```
#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
```

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

$$Z(a, b) = \sin^{2.2}(\sqrt{ab}) - 2\cos^4(1.86b)$$

при $0.15 \leq a \leq 2h_a = 0.18$ $6 \leq b \leq 8h_b = 1$

```
#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

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

$$F(y) = \frac{\sqrt{x^3}y^2 - \cos(y)}{\sin^3(x) - \sin(0.2y)}$$

при $x=0.91253$ $0.6 \leq y \leq 2.5$ $h_b = 0.8$

```
#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

5. **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

6. **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 В[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 В[5,5] = 8

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

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

Двумерный массив В[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

7. **Составить** программу для определения min элемента ниже главной диагонали массива случайных чисел С[8,8]. Поменять местами min элемент и C_{42} .

```
#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 x 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 C(4,2) = " << m[1][3] << L" с элементом C(" << 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 x 8) после смены минимального элемента ниже главной диагонали и C42:" << 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 x 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 $C(4,2) = 67$ с элементом $C(4,6) = 1$

Матрица (8 x 8) после смены минимального элемента ниже главной диагонали и $C(4,2)$:

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

8. **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 $S = \sum_{i=1}^5 \sin^2(c_i)$.
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

```