

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

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

```
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  int main()
7  {
8      setlocale(LC_ALL, "rus");
9      wcout << L"Введите x: " << endl;
10     double x; cin >> x;
11     wcout << L"Введите y: " << endl;
12     double y; cin >> y;
13
14     double inside_cos = M_PI * x + 4 * sqrt(y);
15     double inside_sin = 2 * M_PI - pow(y, 3);
16     double result = 0.252 * pow(cos(inside_cos), 4) + pow(sin(inside_sin), 2);
17
18     wcout << L"Результат вычисления функции Z(x, y) = " << result << endl;
19
20     return 1;|
21 }
22
```

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

```
1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4
5  using namespace std;
6
7  int main()
8  {
9      double step_a = 0.18; double step_b = 1.;
10     double left_a = 0.15; double right_a = 2.;
11     double left_b = 6.; double right_b = 8.;
12     cout << "a\tb\tresult" << endl;
13
14     /* If x is a finite value less than 0, and y is a finite noninteger,
15      a domain error occurs, and a NaN is returned.
16
17      Except as specified below, if x or y is a NaN, the result is a NaN.
18
19      If x is negative, then large negative or positive y values yield a NaN
20      as the function result, with errno set to EDOM, and an invalid
21      (FE_INVALID) floating-point exception. For example, with pow(),
22      one sees this behavior when the absolute value of y is greater than about
23      9.223373e18. */
24
25     for(double a = left_a; a <= right_a; a += step_a)
26     {
27         for(double b = left_b; b <= right_b; b += step_b)
28         {
29             double inside_cos = 1.86 * b;
30             double inside_sin = sqrt(a) * b;
31             double result = pow(abs(sin(inside_sin)), 2.2) - 2. * pow(cos(inside_cos), 4.);
32
33             cout << setprecision(4) << a << '\t' << b << '\t';
34             cout << result << endl;
35         }
36     }
37
38     return 1;
39 }
```

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

```
1  #include <iostream>
2  #include <cmath>
3  #include <fstream>
4
5  using namespace std;
6
7  int main()
8  {
9
10     double x = 0.91253;
11     double result = 0.;
12     double y_left = 0.6;
13     double y_right = 2.4;
14     double y_step = 0.8;
15
16     ofstream file("../result.txt");
17     if(!file.is_open()) return 0;
18
19     cout << "x\ty\tresult" << endl;
20     file << "x\ty\tresult" << endl;
21     for(double y = y_left; y <= y_right; y += y_step)
22     {
23         double dividend = sqrt(pow(x,3)) * y * y - cos(y);
24         double divisor = pow(sin(x), 3) - sin(0.2 * y);
25
26         result = dividend / divisor;
27         cout << x << "\t" << y << "\t" << result << endl;
28         file << x << "\t" << y << "\t" << result << endl;
29     }
30
31     file.close();
32
33     return 1;
34 }
35
```

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.

```
1  #include <iostream>
2  #include <vector>
3  #include <random>
4  #include <algorithm>
5
6  using namespace std;
7
8  int main()
9  {
10     setlocale(LC_ALL, "rus");
11     vector<vector<int>> a;
12     a.resize(6);
13
14     for(int i = 0; i < a.size(); i++)
15         a[i].resize(8);
16
17     std::random_device rd;
18     std::mt19937 gen(rd());
19     std::uniform_int_distribution<> distr(-100, 100); // случайные значения в диапазоне -100, 100
20
21     wcout << L"Двумерный массив A[6,8] случайных чисел:" << endl << endl;
22     int sum = 0;
23     int max_elem = a[0][0];
24     int row_max_elem, col_max_elem;
25     for(int i = 0; i < a.size(); i++)
26     {
27         for(int j = 0; j < a[i].size(); j++)
28         {
29             a[i][j] = distr(gen);
30             cout << a[i][j] << 't';
31             if(a[i][j] > 0)
32             {
33                 sum += a[i][j];
34             }
35
36             if(a[i][j] > max_elem)
37             {
38                 max_elem = a[i][j];
39                 row_max_elem = i;
40                 col_max_elem = j;
41             }
42         }
43         cout << endl;
44     }
45
46     wcout << L"\n\nСумма положительных элементов массива A[6,8] S = " << sum << endl;
47     wcout << L"\nМаксимальный элемент массива = " << max_elem << endl;
48     wcout << L"Номер строки максимального элемента = " << row_max_elem + 1 << endl;
49     wcout << L"Номер столбца максимального элемента = " << col_max_elem + 1 << endl;
50
51     return 1;
52 }
```

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.

```
1  #include <iostream>
2  #include <vector>
3  #include <random>
4  #include <algorithm>
5
6  using namespace std;
7
8  int main()
9  {
10     setlocale(LC_ALL, "rus");
11     vector<vector<int>> b;
12     b.resize(5);
13
14     std::random_device rd;
15     std::mt19937 gen(rd());
16     std::uniform_int_distribution<> distr(1, 100); // случайные значения в диапазоне 1, 100
17
18     wcout << L"Двумерный массив B[5,5] случайных чисел:" << endl << endl;
19     for(int i = 0; i < b.size(); i++)
20     {
21         b[i].resize(5);
22         for(int j = 0; j < b[0].size(); j++)
23         {
24             b[i][j] = distr(gen);
25             cout << b[i][j] << '\t';
26         }
27         cout << endl;
28     }
29
30     cout << endl << endl;
31
32     vector<int> v;
33     vector<int> second_row;
34     vector<int> third_row;
35     for(int i = 0; i < b.size(); i++)
36     {
37         for(int j = 0; j < b[0].size(); j++)
38         {
39             if(i % 2 != 0)
40                 v.push_back(b[i][j]);
41             if(i == 1)
42                 second_row.push_back(b[i][j]);
43             if(i == 2)
44                 third_row.push_back(b[i][j]);
45         }
46     }
47 }
```

```

48     int minimum = *min_element(v.begin(), v.end());
49     wcout << L"Минимальный элемент четных строк массива B[5,5] = " << minimum << endl;
50
51     int min_of_second_row = *min_element(second_row.begin(), second_row.end());
52     vector<int>::iterator iter_min_of_second_row = min_element(second_row.begin(), second_row.end());
53     int pos_min_second_row = distance(second_row.begin(), iter_min_of_second_row);
54     wcout << L"Минимальный элемент второй строки массива B[5,5] = " << min_of_second_row << endl;
55
56     int min_of_third_row = *min_element(third_row.begin(), third_row.end());
57     vector<int>::iterator iter_min_of_third_row = min_element(third_row.begin(), third_row.end());
58     int pos_min_third_row = distance(third_row.begin(), iter_min_of_third_row);
59     wcout << L"Минимальный элемент третьей строки массива B[5,5] = " << min_of_third_row << endl;
60
61
62     int tmp = b[1][pos_min_second_row];
63     b[1][pos_min_second_row] = b[2][pos_min_third_row];
64     b[2][pos_min_third_row] = tmp;
65
66     wcout << L"\nДвумерный массив B[5,5] случайных чисел после перестановки по условию:" << endl << endl;
67     for(int i = 0; i < b.size(); i++)
68     {
69         for(int j = 0; j < b[0].size(); j++)
70             cout << b[i][j] << '\t';
71         cout << endl;
72     }
73
74     return 1;
75 }

```

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

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

```
1  #include <iostream>
2  #include <cmath>
3  #include <vector>
4  #include <random>
5  #include <algorithm>
6  #include <tuple>
7
8  using namespace std;
9
10 int main()
11 {
12     setlocale(LC_ALL, "rus");
13     vector<vector<int>> m;
14     m.resize(8);
15     std::random_device rd;
16     std::mt19937 gen(rd());
17     std::uniform_int_distribution<> distr(1, 100); // случайные значения в диапазоне 1, 100
18
19     // Инициализация матрицы случайными числами
20     vector<tuple<int, int, int>> vt;
21     wcout << L"Матрица (8 x 8) случайных чисел:" << endl;
22     for(int i = 0; i < m.size(); i++)
23     {
24         m[i].resize(8);
25         for(int j = 0; j < m.size(); j++)
26         {
27             m[i][j] = (int)distr(gen);
28             cout << m[i][j] << '\t';
29             if(i > j)
30             {
31                 tuple<int, int, int> t = make_tuple(m[i][j], i + 1, j + 1);
32                 vt.push_back(t);
33             }
34         }
35         cout << endl;
36     }
37     cout << endl;
38
39     // Сохраняю элементы ниже главной диагонали
40     vector<int> v;
41     for(int i = 0; i < vt.size(); i++)
42         v.push_back(get<0>(vt[i]));
43
44     // Поиск минимального элемента ниже главной диагонали
45     int min_elem = *min_element(v.begin(), v.end());
46     vector<int>::iterator it = std::min_element(v.begin(), v.end());
47     int iter = distance(v.begin(), it);
48     int m_i = get<1>(vt[iter]);
49     int m_j = get<2>(vt[iter]);
50     wcout << L"\nМинимальный элемент ниже главной диагонали = " << min_elem << endl;
51
52     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;
53     int tmp = m[1][3];
54     m[1][3] = m[m_i - 1][m_j - 1];
55     m[m_i - 1][m_j - 1] = tmp;
56
57     wcout << L"Матрица (8 x 8) после смены минимального элемента ниже главной диагонали и C42:" << endl;
58     for(int i = 0; i < m.size(); i++)
59     {
60         for(int j = 0; j < m.size(); j++)
61         {
62             cout << m[i][j] << '\t';
63         }
64         cout << endl;
65     }
66     return 1;
67 }
```

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.

```
1  #include <iostream>
2  #include <vector>
3  #include <fstream>
4  #include <random>
5  #include <algorithm>
6
7  using namespace std;
8
9  int main()
10 {
11     setlocale(LC_ALL, "rus");
12     vector<vector<double>> d;
13     d.resize(4);
14
15     std::random_device rd;
16     std::mt19937 gen(rd());
17     std::uniform_real_distribution<> distr(1., 100.); // случайные значения в диапазоне 1, 100
18
19     wcout << L"Двумерный массив D[4,6] случайных вещественных чисел:" << endl << endl;
20     for(int i = 0; i < d.size(); i++)
21     {
22         d[i].resize(6);
23         for(int j = 0; j < d[0].size(); j++)
24         {
25             d[i][j] = distr(gen);
26             cout << d[i][j] << '\t';
27         }
28         cout << endl;
29     }
30
31
32     vector<double> v;
33     for(int i = 0; i < d.size(); i++)
34         for(int j = 0; j < d[0].size(); j++)
35             v.push_back(d[i][j]);
36
37     wcout << L"\nОтсортированные элементы двумерного массива D[4,6]:" << endl << endl;
38     sort(v.begin(), v.end());
39     for(int i = 0; i < v.size(); i++)
40         cout << v[i] << '\t';
41 }
```

```

41
42
43 wcout << L"\n\nМассив C[5] минимальных элементов из D[4,6]:" << endl << endl;
44 vector<double> c;
45 double sum = 0;
46 for(int i = 0; i < 5; i++)
47 {
48     c.push_back(v[i]);
49     cout << c[i] << '\t';
50     sum += pow(sin(c[i]), 2);
51 }
52
53 wcout << L"\n\nСумма элементов массива C[5] по формуле в условии = " << sum;
54
55 double tmp = c.front();
56 c.front() = c.back();
57 c.back() = tmp;
58
59 ofstream file("../result.txt");
60 if(file.is_open())
61 {
62     wcout << L"\n\nМассив C[5] минимальных элементов из D[4,6] после перестановки первого и последнего элемента:" << endl << endl;
63     file << "Массив C[5] минимальных элементов из D[4,6] после перестановки первого и последнего элемента:" << endl << endl;
64     for(int i = 0; i < 5; i++)
65     {
66         cout << c[i] << '\t';
67         file << c[i] << '\t';
68     }
69 }
70
71 file.close();
72 return 1;
73 }

```

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

```