

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курский государственный университет»

Кафедра программного
обеспечения и администрирования
информационных систем

Направление подготовки
математическое обеспечение и
администрирование
информационных систем

Форма обучения очная

Отчет
по лабораторной работе №5
«Стандартная библиотека шаблонов»

Выполнил:

студент группы 213

Тихонов Е.Е.

Проверил:

старший преподаватель кафедры ПОиАИС

Ураева Е.Е.

Курск, 2021

Цель работы: изучить особенности использования стандартной библиотеки шаблонов языка C++.

Задание

Задача 1. Дан массив целых чисел размера n . Перед каждым локальным максимумом (элементы, который строго больше значений соседних с ним ячеек) вставить минимальный по модулю элемент массива или сообщить, что таких элементов нет.

Задача 2. Дано натуральное число n . Используя метод «решета Эратосфена» выписать все простые числа от 1 до n в убывающем порядке.

Задача 3. Даны векторы $v1$ и $v2$, элементами которых являются английские слова, набранные заглавными буквами, причем все слова в каждом векторе различны. Получить вектор v , являющийся внутренним объединением векторов $v1$ и $v2$, в каждой паре которого первое слово начинается с буквы, которой оканчивается второе слово. Порядок следования пар определяется алфавитным порядком первых элементов пар, а для равных первых элементов — порядком вторых элементов, обратным порядку их следования в векторе $v2$. Для построения вектора v выполнить группировку элементов вектора $v2$ по ключу — последней букве слова, используя вариант группировки со вспомогательным мультиотображением m , после чего в цикле по отсортированным элементам вектора $v1$ сформировать требуемое внутреннее объединение, перебирая для каждого элемента вектора $v1$ соответствующие ему элементы отображения m .

Задача 4. Дан вектор v с четным количеством элементов. В первой половине исходного вектора заменить все отрицательные числа на -1 , а во второй — все положительные числа на 1 . Использовать два вызова алгоритма `replace_if` с различными параметрами — функциональными объектами.

Разработка алгоритма

Задача 1

Входные данные: n - целое число, затем n целых чисел подряд.

Выходные данные: ряд целых чисел.

Задача 2

Входные данные: n - целое число.

Выходные данные: ряд простых чисел.

Задача 3

Входные данные: нет.

Выходные данные: некое количество строк, содержащих пары слов.

Задача 4

Входные данные: нет.

Выходные данные: некое количество строк, содержащих данные об объектах программы.

Текст программы

Текст программы для решения задачи 1

```
#include <iostream>
```

```
#include <cmath>
```

```
#include <vector>
```

```
using namespace std;
```

```

int main()
{
    int n, elem, absMin(0);
    cin >> n;
    vector<int> vector;
    for (int i = 0; i < n; i++)
    {
        cin >> elem;
        vector.push_back(elem); //1

        if(i == 0 || abs(elem) < absMin )
        {
            absMin = elem;
        }
    }

    bool LeftMin, RightMin, WriteMessage(true);
    for (std::vector<int>::iterator it = vector.begin(); it !=
vector.end(); ++it) //2
    {
        LeftMin = false;
        RightMin = false;

        if(it == vector.begin())
        {
            LeftMin = true;
        }
        else if(*(it - 1) < *it)
        {
            LeftMin = true;
        }
    }
}

```

```

        if (it == vector.end() - 1)
        {
            RightMin = true;
        }
        else if (*(it + 1) < *it)
        {
            RightMin = true;
        }

        if (LeftMin && RightMin)
        {
            WriteMessage = false;
            it = vector.insert(it, absMin); //3
            it++;
        }
    }

    if (WriteMessage)
    {
        cout << "Hasn't local maximums";
    }
    else
    {
        for (std::vector<int>::iterator it = vector.begin(); it !=
vector.end(); ++it)
        {
            cout << *it << " ";
        }
    }
    vector.clear(); //4

    return 0;

```

```
}
```

Текст программы для решения задачи 2

```
#include <iostream>
```

```
#include <cmath>
```

```
#include <set>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int n;
```

```
    cin >> n;
```

```
    set<int> set;
```

```
    for (int i = 1; i <= n; i++)
```

```
    {
```

```
        set.insert(i); //1
```

```
    }
```

```
    std::set<int>::iterator it = set.begin();//2
```

```
    while(++it != set.end())//3
```

```
    {
```

```
        std::set<int>::iterator it2 = it;
```

```
        it2++;
```

```
        while(it2 != set.end())
```

```
        {
```

```
            if(*it2%*it==0)
```

```
            {
```

```
                it2 = set.erase(it2);//4
```

```
            }
```

```
        else
```

```
        {
```

```
            it2++;
```

```
        }
```

```
    }
```

```

    }

    copy(set.rbegin(), set.rend(), ostream_iterator<int>(cout, "
")); //5 6
    set.clear(); //7

    return 0;
}

```

Текст программы для решения задачи 3

```

#include <iostream>
#include <map>
#include <vector>
#include <set>
#include <string>

using namespace std;

int main()
{
    set<string> v1 = { "BBB", "BCB", "BZA", "ABA", "AAD", "DAD", "CCC",
"CAA" };
    vector<string> v2 = { "CACD", "BBA", "BAA", "AFAB", "DD", "ZZB",
"ADBZ", "DZ" };

    multimap<char, string> m;
    for (int i = v2.size()-1; i >= 0; i--)
    {
        m.insert(pair<char, string>(v2[i][v2[i].size()-1],
v2[i])); //1
    }

    vector<vector<string>> v;

```

```

for (auto it = v1.begin(); it != v1.end(); it++)
{
    char c = (*it)[0];

    auto r = m.equal_range(c);//2
    for (auto it2 = r.first; it2 != r.second; it2++)
    {
        v.push_back(vector<string>({ *it, it2->second }));
    }
}

m.clear();//3

for (size_t i = 0; i < v.size(); i++)
{
    for (size_t j = 0; j < 2; j++)
    {
        cout << v[i][j] << " ";
    }
    cout << endl;
}

return 0;
}

```

Текст программы для решения задачи 4

```

#include <iostream>
#include <algorithm>
#include <vector>
#include <string>

using namespace std;

```



```

class Worker
{
private:

    string Name;
    int ReportCard;
    float PricePerHour;
    int WorkedHours;

    //property getters
public:
    string getName()
    {
        return this->Name;
    }
    int getReportCard()
    {
        return this->ReportCard;
    }
    float getPricePerHour()
    {
        return this->PricePerHour;
    }
    int getWorkedHours()
    {
        return this->WorkedHours;
    }

    //property setters
public:
    void setName(string Name)
    {

```

```

        this->Name = Name;
    }
    void setReportCard(int ReportCard)
    {
        this->ReportCard = ReportCard;
    }
    void setPricePerHour(float PricePerHour)
    {
        this->PricePerHour = PricePerHour;
    }
    void setWorkedHours(int WorkedHours)
    {
        this->WorkedHours = WorkedHours;
    }

```

public:

```

    Worker(string Name, int ReportCard, float PricePerHour = 100, int
        WorkedHours = 10)
    {
        setName(Name);
        setReportCard(ReportCard);
        setPricePerHour(PricePerHour);
        setWorkedHours(WorkedHours);
    }

    void Print()
    {
        cout << this->Name << " ";
        cout << this->ReportCard << " ";
        cout << to_string(this->PricePerHour) << " ";
        cout << to_string(this->WorkedHours) << endl;
    }

```

```

    Worker operator = (int I)
    {
        this->ReportCard = I;
        return *this;
    }
};

int main()
{
    vector<Worker> workers = {
        Worker("A", -100, 130, -15),
        Worker("B", 120, 140, 10),
        Worker("C", -100, -10, 30),
        Worker("D", -10, 14, 170),
        Worker("E", 320, -20, -5),
        Worker("F", 4, 23, 60),
        Worker("G", -3, 12, 73),
        Worker("H", -333, 6, 8)
    };

    replace_if(workers.begin(),
workers.begin()+workers.size()/2,      [&](Worker      x)      {return
x.getReportCard() < 0; }, -1);
    replace_if(workers.begin() + workers.size() / 2, workers.end(),
[&](Worker& x) {return x.getReportCard() > 0; }, 1);

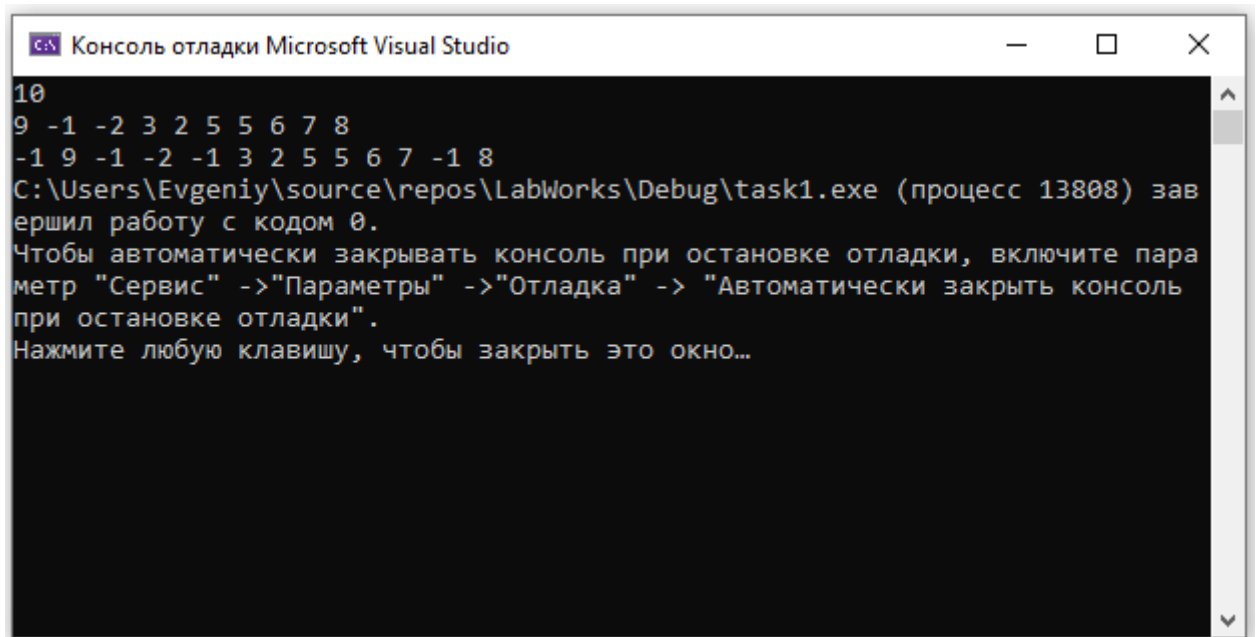
    for (auto x : workers) {
        x.Print();
    }

    return 0;
}

```

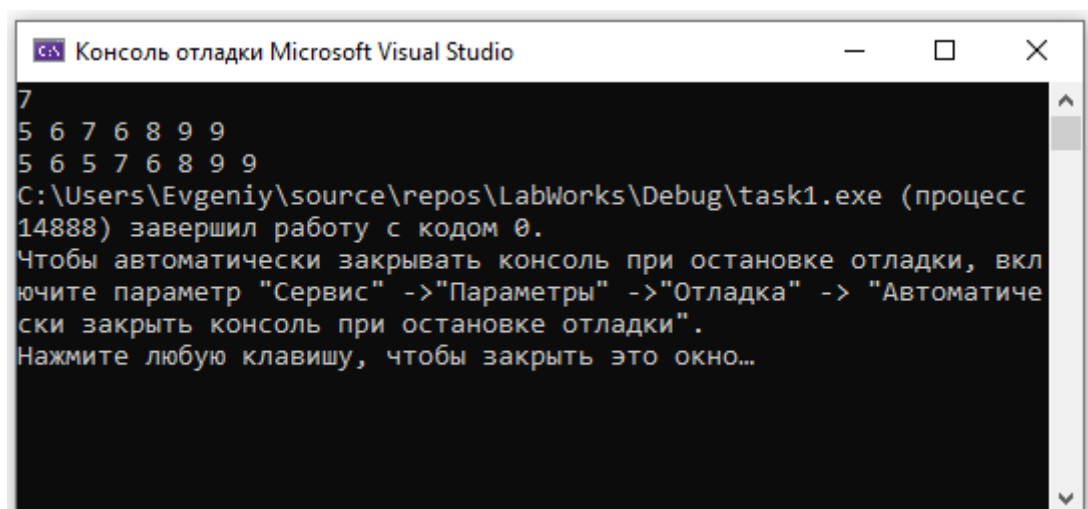
Тестирование программы

Тестирование задачи 1 представлено на рисунках 1, 2.



```
Консоль отладки Microsoft Visual Studio
10
9 -1 -2 3 2 5 5 6 7 8
-1 9 -1 -2 -1 3 2 5 5 6 7 -1 8
C:\Users\Evgeniy\source\repos\LabWorks\Debug\task1.exe (процесс 13808) за-
вершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите пара-
метр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль
при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

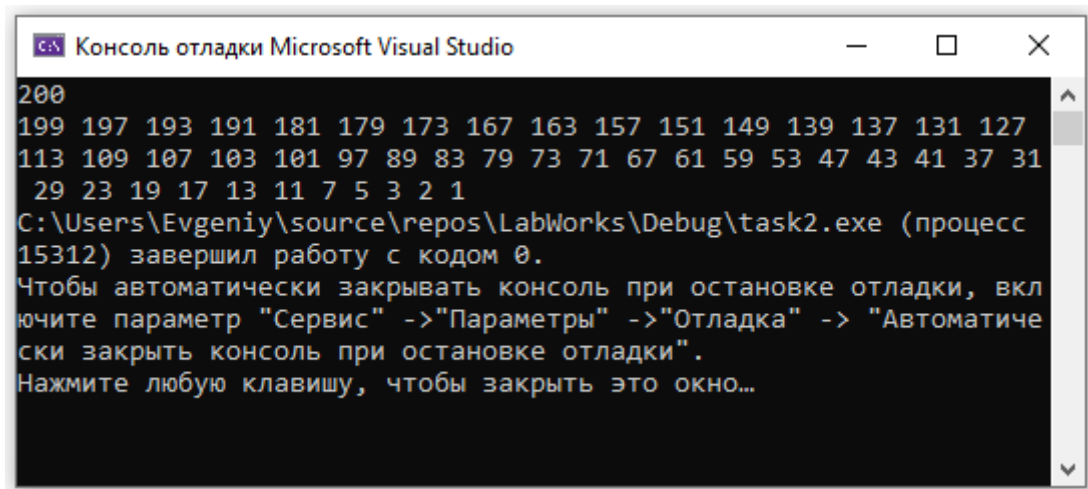
Рисунок 1 - Тест 1 задачи 1



```
Консоль отладки Microsoft Visual Studio
7
5 6 7 6 8 9 9
5 6 5 7 6 8 9 9
C:\Users\Evgeniy\source\repos\LabWorks\Debug\task1.exe (процесс
14888) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, вкл-
ючите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматиче-
ски закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

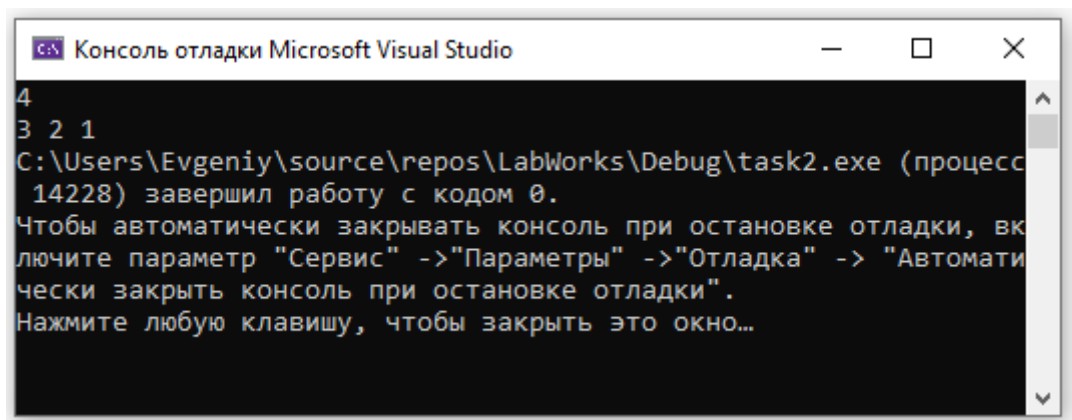
Рисунок 2 - Тест 2 задачи 1

Тестирование задачи 2 представлено на рисунках 3, 4.



```
Консоль отладки Microsoft Visual Studio
200
199 197 193 191 181 179 173 167 163 157 151 149 139 137 131 127
113 109 107 103 101 97 89 83 79 73 71 67 61 59 53 47 43 41 37 31
29 23 19 17 13 11 7 5 3 2 1
C:\Users\Evgeniy\source\repos\LabWorks\Debug\task2.exe (процесс
15312) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, вкл
ючите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматиче
ски закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

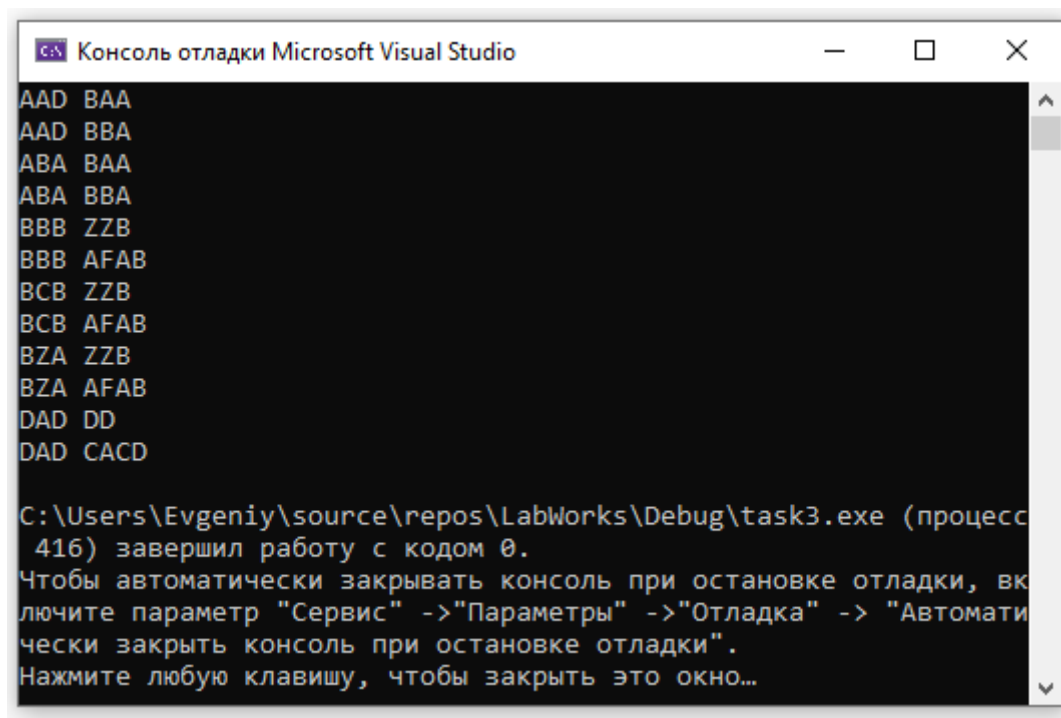
Рисунок 3 - Тест 1 задачи 2



```
Консоль отладки Microsoft Visual Studio
4
3 2 1
C:\Users\Evgeniy\source\repos\LabWorks\Debug\task2.exe (процесс
14228) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, вк
лючите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автомати
чески закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 4 - Тест 2 задачи 2

Тестирование задачи 3 представлено на рисунках 5, 6.

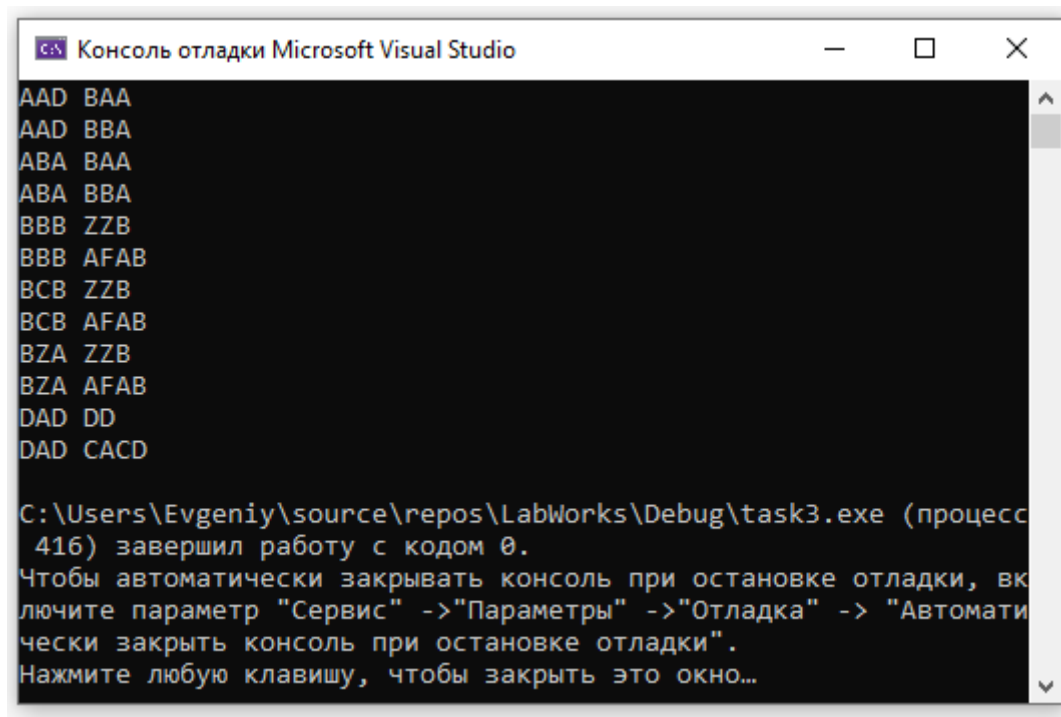


```
Консоль отладки Microsoft Visual Studio

AAD BAA
AAD BBA
ABA BAA
ABA BBA
BBB ZZB
BBB AFAB
BCB ZZB
BCB AFAB
BZA ZZB
BZA AFAB
DAD DD
DAD CACD

C:\Users\Evgeniy\source\repos\LabWorks\Debug\task3.exe (процесс 416) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 5 - Тест 1 задачи 3



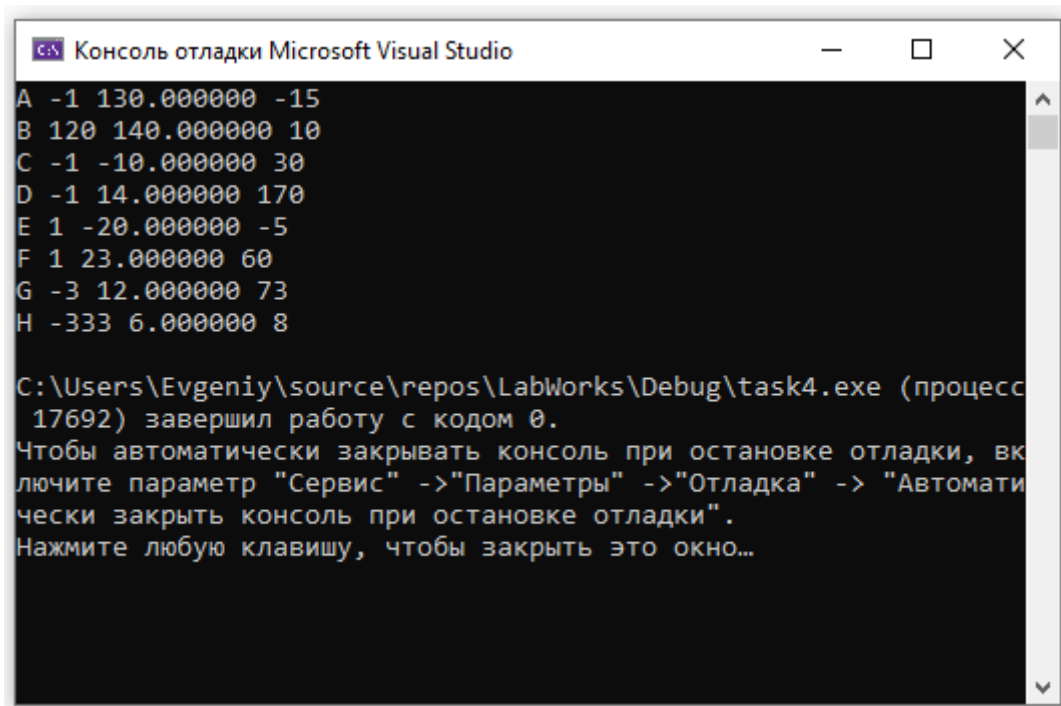
```
Консоль отладки Microsoft Visual Studio

AAD BAA
AAD BBA
ABA BAA
ABA BBA
BBB ZZB
BBB AFAB
BCB ZZB
BCB AFAB
BZA ZZB
BZA AFAB
DAD DD
DAD CACD

C:\Users\Evgeniy\source\repos\LabWorks\Debug\task3.exe (процесс 416) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрывать консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 6 - Тест 2 задачи 3

Тестирование задачи 4 представлено на рисунках 7, 8.

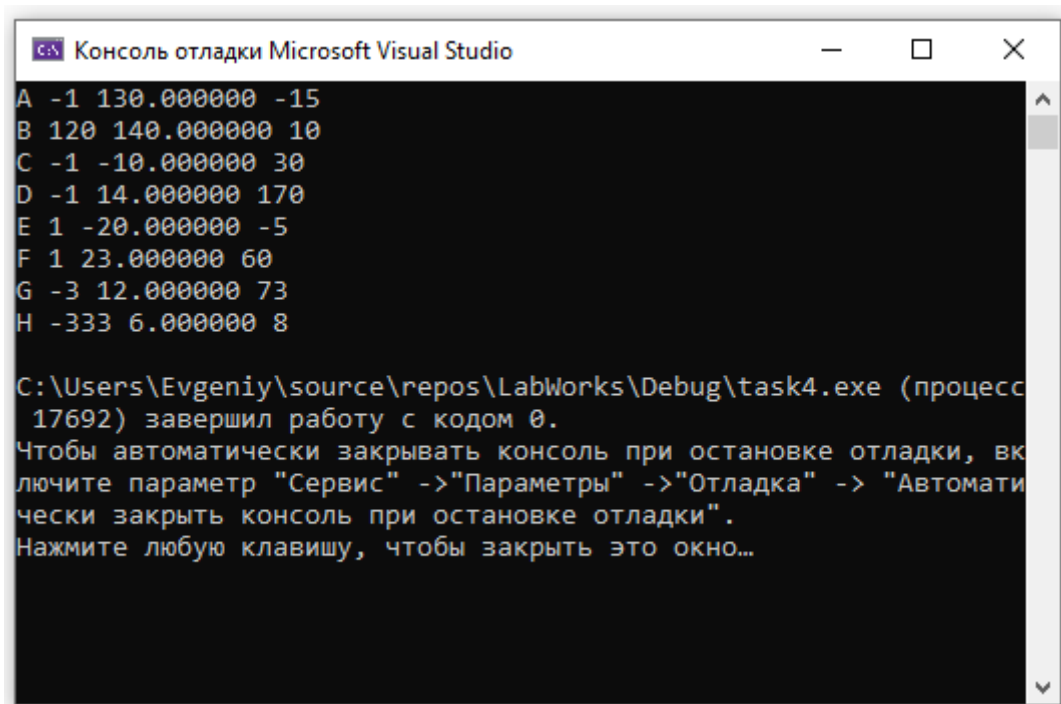


```
Консоль отладки Microsoft Visual Studio

A -1 130.000000 -15
B 120 140.000000 10
C -1 -10.000000 30
D -1 14.000000 170
E 1 -20.000000 -5
F 1 23.000000 60
G -3 12.000000 73
H -333 6.000000 8

C:\Users\Evgeniy\source\repos\LabWorks\Debug\task4.exe (процесс 17692) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 7 - Тест 1 задачи 4



```
Консоль отладки Microsoft Visual Studio

A -1 130.000000 -15
B 120 140.000000 10
C -1 -10.000000 30
D -1 14.000000 170
E 1 -20.000000 -5
F 1 23.000000 60
G -3 12.000000 73
H -333 6.000000 8

C:\Users\Evgeniy\source\repos\LabWorks\Debug\task4.exe (процесс 17692) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рисунок 8 - Тест 2 задачи 4