

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Департамент программной инженерии

Пояснительная записка к домашнему заданию
по дисциплине «Архитектура вычислительных систем»
Параллельное программирование. Взаимодействие и синхронизация.

Исполнитель:
студент группы
БПИ193(1)
Полянская П. А.
Вариант 17
13.12.2020 г.

Текст задачи.

17. Задача о нелюдимых садовниках. Имеется пустой участок земли (двумерный массив) и план сада, который необходимо реализовать. Эту задачу выполняют два садовника, которые не хотят встречаться друг с другом. Первый садовник начинает работу с верхнего левого угла сада и перемещается слева направо, сделав ряд, он спускается вниз. Второй садовник начинает работу с нижнего правого угла сада и перемещается снизу вверх, сделав ряд, он перемещается влево. Если садовник видит, что участок сада уже выполнен другим садовником, он идет дальше. Садовники должны работать параллельно. Создать многопоточное приложение, моделирующее работу садовников. При решении задачи использовать мутексы.

О программе.

Используется мутексы (заданы по условию). Поток запрашивает монопольное использование общих данных, защищаемых мьютексом. В данной задаче это вектор, содержащий пару целых чисел – координаты на часть участка, на которых уже был какой-либо садовник.

Создаются два потока – два садовника.

Используемые источники.

1) Mutex. [Электронный ресурс]

<https://medium.com/nuances-of-programming/c-мьютекс-пишем-наш-первый-код-для-многопоточной-среды-543a3d60ef30> (дата обращения: 13.12.2020)

2) SoftCraft, сайт по учебной дисциплине. [Электронный ресурс]

<http://softcraft.ru/> (дата обращения: 13.12.2020)

Пояснения по решению.

Пояснения присутствуют в коде, и есть комментарии о роли каждого метода.

Тестирование различных входных данных.

1) Входные данные правильные.

```
You can write down only int numbers from 1 to 100 ([1,100]).
Write down a number of rows.
4
Write down a number of columns.
2
Gardener, who steps in a vertical way, was in spot of bed (3;1).
Gardener, who steps in a horizontal way, was in spot of bed (0;0).
Gardener, who steps in a vertical way, was in spot of bed (2;1).
Gardener, who steps in a horizontal way, was in spot of bed (0;1).
Gardener, who steps in a vertical way, was in spot of bed (1;1).
Gardener, who steps in a vertical way, was in spot of bed (3;0).
Gardener, who steps in a vertical way, was in spot of bed (2;0).
Gardener, who steps in a vertical way, was in spot of bed (1;0).
```

2) Введены неправильные входные данные.

Были попробованы введены строка(буквы), дробное число, отрицательное число, число равное 0 и больше 100.

Все неправильные значения были обработаны и выведена строка о неправильном введённом значении.

Пользователь вводит число, пока оно не окажется правильным.

```
You can write down only int numbers from 1 to 100 ([1,100]).
Write down a number of rows.
-1
It's a wrong input. Try again.
пнгшрщ
It's a wrong input. Try again.
5,2
It's a wrong input. Try again.
5.2
It's a wrong input. Try again.
-
It's a wrong input. Try again.
101
It's a wrong input. Try again.
3
Write down a number of columns.
2
Gardener, who steps in a vertical way, was in spot of bed (2;1).
Gardener, who steps in a horizontal way, was in spot of bed (0;0).
Gardener, who steps in a vertical way, was in spot of bed (1;1).
Gardener, who steps in a vertical way, was in spot of bed (0;1).
Gardener, who steps in a vertical way, was in spot of bed (2;0).
Gardener, who steps in a vertical way, was in spot of bed (1;0).
```

3)

```
You can write down only int numbers from 1 to 100 ([1,100]).
Write down a number of rows.
5
Write down a number of columns.
5
Gardener, who steps in a vertical way, was in spot of bed (4;4).
Gardener, who steps in a horizontal way, was in spot of bed (0;0).
Gardener, who steps in a vertical way, was in spot of bed (3;4).
Gardener, who steps in a horizontal way, was in spot of bed (0;1).
Gardener, who steps in a vertical way, was in spot of bed (2;4).
Gardener, who steps in a vertical way, was in spot of bed (1;4).
Gardener, who steps in a horizontal way, was in spot of bed (0;2).
Gardener, who steps in a vertical way, was in spot of bed (0;4).
Gardener, who steps in a horizontal way, was in spot of bed (0;3).
Gardener, who steps in a vertical way, was in spot of bed (4;3).
Gardener, who steps in a vertical way, was in spot of bed (3;3).
Gardener, who steps in a vertical way, was in spot of bed (2;3).
Gardener, who steps in a vertical way, was in spot of bed (1;3).
Gardener, who steps in a vertical way, was in spot of bed (4;2).
Gardener, who steps in a vertical way, was in spot of bed (3;2).
Gardener, who steps in a vertical way, was in spot of bed (2;2).
Gardener, who steps in a vertical way, was in spot of bed (1;2).
Gardener, who steps in a vertical way, was in spot of bed (4;1).
Gardener, who steps in a vertical way, was in spot of bed (3;1).
Gardener, who steps in a vertical way, was in spot of bed (2;1).
Gardener, who steps in a vertical way, was in spot of bed (1;1).
Gardener, who steps in a vertical way, was in spot of bed (4;0).
Gardener, who steps in a vertical way, was in spot of bed (3;0).
Gardener, who steps in a vertical way, was in spot of bed (2;0).
Gardener, who steps in a vertical way, was in spot of bed (1;0).
```

4)

```

You can write down only int numbers from 1 to 100 ([1,100]).
Write down a number of rows.
2
Write down a number of columns.
2
Gardener, who steps in a horizontal way, was in spot of bed (0;0).
Gardener, who steps in a vertical way, was in spot of bed (1;1).
Gardener, who steps in a horizontal way, was in spot of bed (0;1).
Gardener, who steps in a horizontal way, was in spot of bed (1;0).

```

Исходный код.

```

#include <iostream>
#include <fstream>
#include <cstdlib>
#include <stdlib.h>
#include <string>
#include <vector>
#include <thread>
#include <mutex>
using namespace std;

//Полянская Полина БПИ193 17 вариант

mutex g_lock;
//вектор с координатами проработанных частей участка
static vector<pair<int, int>> Garden;
//число строк в участке
static int rows = 0;
//число столбцов в участке
static int columns = 0;

/// <summary>
/// Проверяет, был ли какой-либо садовник в данной части участка.
/// Если не был, то садовник начинает работать на данной части участка.
/// </summary>
/// <param name="i">позиция в участке по столбцам</param>
/// <param name="j">позиция в участке по строкам</param>
/// <param name="way">информация о том, как ходит садовник</param>
void Adder(int i, int j, string way)
{
    g_lock.lock();
    bool checker = false;
    auto it = Garden.begin();
    //Проверка на то, есть ли такая координата в списке проработанных частях участка
    while (it != Garden.end())
    {
        if (it->first == i && it->second == j)
        {
            checker = true;
        }
        it++;
    }
    //Если такой координаты не было, то садовник приступает к работе.
    if (!checker)
    {
        Garden.push_back(make_pair(i, j));
        cout << "Gardener" << way << " was in spot of bed (" << i << "; " << j <<
        ")." << endl;
    }
    g_lock.unlock();
}

/// <summary>

```

```

/// Метод проходит по всему участку по горизонтальному пути.
/// </summary>
void fillGarden1()
{
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            Adder(i, j, " , who steps in a horizontal way, ");
        }
    }
}

/// <summary>
/// Метод проходит по всему участку вертикальному пути.
/// </summary>
void fillGarden2()
{
    for (int j = columns - 1; j >= 0; j--)
    {
        for (int i = rows - 1; i >= 0; i--)
        {
            Adder(i, j, " , who steps in a vertical way, ");
        }
    }
}

/// <summary>
/// Превращает строку в число.
/// </summary>
/// <param name="str">входящая строка</param>
/// <returns></returns>
int Converter(string str)
{
    cout << str << endl;
    bool Bool = true;
    string check;
    int res = 0;
    while (Bool)
    {
        try {
            cin >> check;
            res = stoi(check);
            //проверяет если число <=0, больше 100 или не целое, то повторяется
            if (res <= 0 || to_string(res) != check || res > 100)
            {
                throw exception();
            }
            Bool = false;
        }
        catch (exception e) {
            cout << "It's a wrong input. Try again." << endl;
        }
    }
    return res;
}

int main()
{
    cout << "You can write down only int numbers from 1 to 100 ([1,100])." << endl;
    rows = Converter("Write down a number of rows.");
    columns = Converter("Write down a number of columns.");
    //создаются два потока - два садовника
    thread* thread1 = new thread{ fillGarden1 };

```

```
    thread* thread2 = new thread{ fillGarden2 };  
    thread1->join();  
    thread2->join();  
    delete thread1;  
    delete thread2;  
}
```