**Лабораторная работа №6**

**«Программирование алгоритмов формирования и обработки вложенных последовательностей»**

**Задача:** Сформируйте вложенную последовательность K [10,12] по правилу:

Где r – случайное число из отрезка [0,1]. Определите в каждом столбце количество простых чисел и запишите его в соответствующий элемент последовательности L. Выведите последовательность L

Примечание:

1. Python: Использовать функции и модули.

В основной части программы происходит ввод строк и столбцов и вызов Module2 содержащий в себе функции: input\_mas,out\_mas, input\_out\_simple.

А) Написать модуль (Module), содержащий функцию IsPrime которая выполняла бы проверку числа на простоту.

Б) Написать модуль (Module2), содержащий следующие функции:

input\_mas – данная функция должна формировать случайную вложенную последовательность

out\_mas – данная функция должна выводить случайную вложенную последовательность

input\_out\_simple – данная функция должна формировать и выводить последовательность L

Результат представить:

В виде вывода таблицы, и последовательности L.

В основной части программы происходит ввод строк и столбцов и вызов функции outmas содержащую в себе DLL библиотеку.

1. С++: Использовать функции и DLL библиотеку

В основной части программы происходит ввод строк и столбцов.

А) Написать DLL библиотеку, содержащий внутри себя следующие функции:

memory\_arr – данная функция должна формировать случайную матрицу

memory\_L – данная функция должна формировать последовательность L

is\_prime – данная функция должна являться булевой, выполняя проверку числа на простоту.

Б) Написать функцию outmas в которой будет использоваться DLL библиотека написанная ранее, а также в данной функции должно выполняться выделения динамической памяти (в дальнейшем её чистка) и выполняться вывод матрицы в dataGridView1 и последовательности L в dataGridView2.

Результат представить:

В виде вывода таблицы, и последовательности L.

**Алгоритм решения задачи:**



**Рис.1.** Общая блок-схема решения задачи



**Рис.2.** Алгоритм модуля (Module)



**Рис.3.** Общая блок-схема модуля (Module2)



**Рис.4.** Алгоритм функции input\_mas



**Рис.5.** Алгоритм функции out\_mas



**Рис.6.** Алгоритм функции input\_out\_simple

**Код программы :**

import Module2

k = int(input('Введите количество столбцов : '))

m = int(input('Введите количество строк : '))

matrix=Module2.input\_mas(k,m)

Module2.out\_mas(matrix,k,m)

simple=Module2.input\_out\_simple(matrix,k,m)

exit()

**Код модуля (Module):**

def IsPrime(n):

d = 2

while n % d != 0:

d += 1

return d == n

**Код модуля (Module2):**

import random

import Module

def input\_mas(k,m):

for i in range(k):

for j in range(m):

matrix=[[i+j for i in random.sample(range(10),m)]

for j in random.sample(range(10),k)]

return matrix

def input\_out\_simple(matrix,k,m):

simple = []

cumsum=0

for i in range(m):

for j in range(k):

if Module.IsPrime(matrix[j][i]):

cumsum=cumsum+1

simple.append(cumsum)

cumsum=0

print('\nПоследовательность L : ',simple)

return matrix

simple.clear()

def out\_mas(matrix,k,m):

print('\nТаблица :\n')

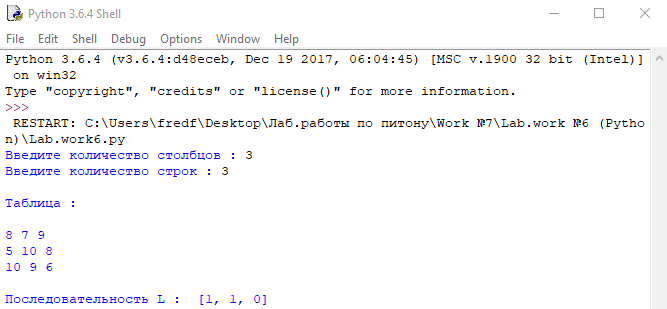
for i in range(k):

for j in range(m):

print(matrix[i][j],end=' ')

print(" ")

**Результат работы программы:**



**Рис.7.** Результат работы приложения

**C++**

**Код программы (Windows Form):**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include "MathFuncsDll.h"

.

.

.

private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) {

int n, m;

n = Convert::ToDouble(textBox1->Text);

m = Convert::ToDouble(textBox2->Text);

outmas(n, m);

}

void outmas(int n, int m)

{

dataGridView1->RowCount = n;

dataGridView1->ColumnCount = m;

srand(time(NULL));

int \*\*arr = new int\*[n];

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

{

arr[i] = new int[m];

}

MathFuncs::MyMathFuncs::memory\_arr(n, arr, m);

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

{

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

{

dataGridView1->Rows[i]->Cells[j]->Value = arr[i][j];

}

}

dataGridView2->ColumnCount = m;

int \*L = new int[m];

MathFuncs::MyMathFuncs::memory\_L(n,arr,m,L);

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

{

dataGridView2->Rows[0]->Cells[i]->Value = L[i];

}

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

{

delete[](arr[i]);

}

delete[]arr;

}

};

}

**Код библиотеки (MathFuncsDll.cpp):**

#include "stdafx.h"

#include "MathFuncsDll.h"

#include <stdexcept>

using namespace std;

bool is\_prime(int n)

{

if ((n == 0) || (n == 1))

{

return false;

}

for (int i = 2; i <= n / 2; i++)

{

if (n % i == 0)

return false;

}

return true;

}

namespace MathFuncs

{

void MyMathFuncs::memory\_arr(int n, int \*\*arr, int m)

{

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

{

arr[i] = new int[m];

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

{

arr[i][j] = rand() % 10;

}

}

}

void MyMathFuncs::memory\_L(int n, int \*\*arr, int m, int \*L)

{

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

{

L[i] = 0;

for (int j = 0; j < n; j++)

{

if (is\_prime(arr[j][i]))

{

L[i]++;

}

}

}

}

}

**Код заголовочного файла (MathFuncsDll.h):**

#pragma once

#ifdef MATHFUNCSDLL\_EXPORTS

#define MATHFUNCSDLL\_API \_\_declspec(dllexport)

#else

#define MATHFUNCSDLL\_API \_\_declspec(dllimport)

#endif

namespace MathFuncs

{

class MyMathFuncs

{

public:

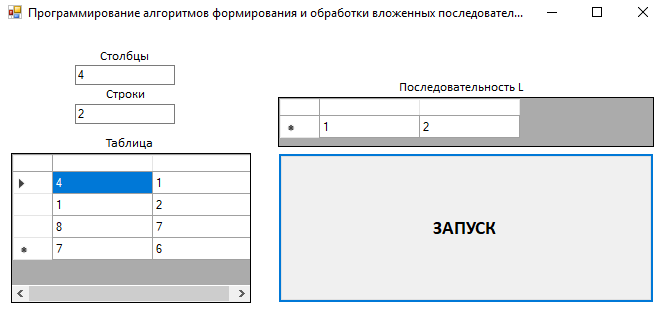
static MATHFUNCSDLL\_API void memory\_arr(int n, int \*\*arr, int m);

static MATHFUNCSDLL\_API void memory\_L(int n, int \*\*arr, int m,int \*L);

};

}

**Результат работы программы:**



**Рис.1.** Результат работы Windows Form