**Федеральное государственное автономное образовательное учреждение**

**высшего профессионального образования**

**"Национальный исследовательский университет**

**"Высшая школа экономики"**

Московский институт электроники и математики НИУ ВШЭ  
Департамент компьютерной инженерии

**Курс: Информатика и программирование**

**ОТЧЕТ**  
**по лабораторной работе № 2**

|  |  |  |  |
| --- | --- | --- | --- |
| **Раздел** | **Max оценка** | **Итог.**  **Оценка**  **1** | **Итог.**  **Оценка**  **2** |
| Тесты | 1 |  |  |
| Работа программы | 1 |  |  |
| Алгоритм | 3 |  |  |
| Вопросы | 2 |  |  |
| Доп. Задание | 3 |  |  |

**Студент: Жалкова Наталия Евгеньевна**  
**Группа: БИВ153, 1 подгруппа**   
**Вариант: 76**  
**Руководитель: Хруслова Диана Владимировна  
Оценка:**

**МОСКВА 2016**

# Содержание

[Содержание 1](#_Toc443325017)

[Условия задач 2](#_Toc443325018)

[Распечатка программы 3](#_Toc443325019)

[Тестовые примеры 8](#_Toc443325020)

# Условия задач

В заданной целочисленной прямоугольной матрице поменять местами следующие два элемента:

Первый положительный и последний четный.

При выполнении этой лабораторной работы разрабатываются две программы. В первой обращение к элементам матрицы происходит при помощи индекса. Во второй – по указателю. В первой части все вычисления, ввод и вывод происходят в главной функции. Во второй части ввод данных, вывод результата и вычисления оформляются как три отдельных функции.

# Распечатка программы

**/\*1 задание\*/**

#include <stdio.h>

using namespace std;

int main (){

short a[10][10], n, m, i, j, c, i1=-1, j1=-1, i2=-1, j2=-1;

float rowr;

printf("\r\nHello, my dear friend!\r\n");

////ВВОД ДЛИНЫ

printf("Please, enter size of NxM matrix, 0<N<11, 0<M<11\r\n");

do

{

scanf("%f",&rowr);

if (rowr!=(short)rowr||rowr<=0||rowr>10)

printf("Enter integer number 0<N<11\r\n");

}

while (rowr!=(short)rowr||rowr<=0||rowr>10); //проверка

n=(short)rowr;

do

{

scanf("%f",&rowr);

if (rowr!=(short)rowr||rowr<=0||rowr>10)

printf("Enter integer number 0<M<11\r\n");

}

while (rowr!=(short)rowr||rowr<=0||rowr>10); //проверка

m=(short)rowr;

////ВВОД МАТРИЦЫ

printf("Enter integer element of matrix %dx%d\r\n",n,m);

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

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

{do

{scanf("%f",&rowr);

if(rowr!=(short)rowr)

printf("Print integer element!\r\n");

}

while(rowr!=(short)rowr); //проверка

a[i][j]=(short)rowr;

}

////ВЫВОД МАТРИЦЫ

printf("Your martix:\r\n");

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

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

printf("%4d",a[i][j]);

printf("\r\n");

}

////ВЫЧИСЛЕНИЯ

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

for (j=0;j<m&&i1==-1;j++) //идем по циклу пока не нашли первый

if(a[i][j]>0) //нашли первый положительный

{i1=i;

j1=j;

}

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

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

if(!(a[i][j]%2)) //нашли последний четный

{i2=i;

j2=j;

}

if (i1!=-1&&i2!=-1) //если оба найдены производится перестановка

{c=a[i1][j1];

a[i1][j1]=a[i2][j2];

a[i2][j2]=c;}

////ВЫВОД МАТРИЦЫ

if (i1<0)

printf("Positive element NOT FOUND\r\n"); //положительный не найден

if (i2<0)

printf("Even element NOT FOUND\r\n"); //четный не найден

if ((i1==i2&&j1==j2&&i1!=-1)||(a[i1][j1]==a[i2][j2]&&i1!=-1))

printf("First positive element = last even element\r\n"); //первый положительный он же последний четный

else

if (i1!=-1&&i2!=-1)

{

printf("Your new martix:\r\n");

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

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

printf("%4d",a[i][j]);

printf("\r\n");

}

}

return 0;}

**/\*2 задание\*/**

#include <stdio.h>

using namespace std;

////ВВОД

void vvod (short a[][10],short \*n,short \*m){

short i,\*uj;

float rowr;

printf("Please, enter size of NxM matrix, 0<N<11, 0<M<11\r\n");

do

{

scanf("%f",&rowr);

if (rowr!=(short)rowr||rowr<=0||rowr>10)

printf("Enter integer number 0<N<11\r\n");

}

while (rowr!=(short)rowr||rowr<=0||rowr>10); //проверка

\*n=(short)rowr;

do

{

scanf("%f",&rowr);

if (rowr!=(short)rowr||rowr<=0||rowr>10)

printf("Enter integer number 0<M<11\r\n");

}

while (rowr!=(short)rowr||rowr<=0||rowr>10); //проверка

\*m=(short)rowr;

printf("Enter integer element of matrix %dx%d\r\n",\*n,\*m);

for (i=0;i<\*n;i++)

for (uj=a[i];uj<a[i]+\*m;uj++)

{do

{scanf("%f",&rowr);

if(rowr!=(short)rowr)

printf("Print integer element!\r\n");

}

while(rowr!=(short)rowr); //проверка

\*uj=(short)rowr;

}

}

////ВЫВОД

void vivod (short a[][10],short n,short m){

short i,\*uj;

for (i=0;i<n;i++) //строки

{for (uj=a[i];uj<a[i]+m;uj++)

printf("%4d",\*uj);

printf("\r\n");

}

}

////ВЫЧИСЛЕНИЯ

void count (short a[][10],short n,short m, short \*\*u1, short \*\*u2){

short i,\*j;

\*u1=\*u2=NULL;

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

for (j=a[i];j<a[i]+m&&\*u1==NULL;j++) //идем по циклу пока не нашли первый

if (\*j>0) //нашли первый положительный

\*u1=j;

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

for (j=a[i];j<a[i]+m;j++)

if (!(\*j%2)) //нашли последний четный

\*u2=j;

}

////ПЕРЕСТАНОВКА

void perest (short \*u1, short \*u2){

short c;

c=\*u1;

\*u1=\*u2;

\*u2=c;

}

////ГЛАВНАЯ ПРОГРАММА

int main (){

short a[10][10],n,m,\*u1,\*u2;

printf("\r\nHello, my dear friend!\r\n");

vvod(a,&n,&m);

printf("Your martix:\r\n");

vivod(a,n,m);

count(a,n,m,&u1,&u2);

if(u1&&u2&&\*u1!=\*u2) //если указатели существуют и это не одно и то же число

{perest(u1,u2);

printf("Your new martix:\r\n");

vivod(a,n,m);

}

if (u1==NULL)

printf("Positive element NOT FOUND\r\n");

else

if (u2==NULL)

printf("Even element NOT FOUND\r\n");

else

if (\*u1==\*u2)

printf("First positive element = last even element\r\n");

return 0;

}

# Тестовые примеры

|  |  |  |
| --- | --- | --- |
| **№** | **Исходные данные** | **Результат** |
| **1** |  | Positive element NOT FOUND  Even element NOT FOUND |
| **2** |  | Positive element NOT FOUND |
| **3** |  | Even element NOT FOUND |
| **4** |  | First positive element = last even element |
| **5** |  | First positive element = last even element |
| **6** |  |  |