Министерство Образования, Культуры и Исследований

**Молдавский Государственный Университет**

**Факультет Математики и Информатики**

**Департамент Информатики**

ОТЧЕТ

по предмету

Технологии программирования

Выполнил:

студент группы IA2003

Ceban Mihail

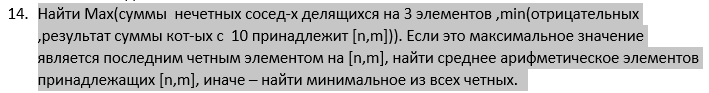
Кишинев, 2020

Содержание отчета

* Lab 1. Формирование и преобразование элементов одномерного массива.
* Lab 2. Обработка элементов одномерного массива.
* Lab 3. [Подсчет и поиск элементов одномерного массива.](https://moodle.usm.md/mod/resource/view.php?id=81817)
* Lab 4. Вложенные циклы в матричных задачах.
* Lab 5. Поиск элементов в матрице, преобразование матрицы.
* Lab 6. Деление двумерного массива (Матрицы (М) ) на зоны : I,II,III,IV и N("северная"),S("южная"),E("западная"),V("восточная").
* Lab 7. Функции , имеющие тип и возвращающие значения
* Lab 8. Ввод ,вывод и обработка элементов ОМ с помощью функций
* Lab 9. Ввод  и обработка двумерного массива с использованием функций
* Lab 10. Указатели. Использование указателей при передаче нескольких значений из функции в основную функцию программы
* Lab 11. Указатели.Использование указателей при передаче элементов массива.
* Lab 12. Обработка последовательностей символов.Тип данных CHAR.
* Lab 13. Обработка последовательностей символов.Тип данных CHAR(используя функции из библиотеки <STRING.H>).
* Lab 14. Тип данных СТРУКТУРА. Объявление переменной типа СТРУКТУРЫ,ввод и вывод данных структуры. Массивы структур.
* Lab 15. Создание проектов : статической библиотеки и консольного приложения .
* Lab 16. Рекурсия . Использование рекурсивных функции для ввода/вывода элементов массивов, для обработки элементов массивов (сумма, произведение, среднее арифм.).
* Lab 17. Текстовые файлы. Операции над файлами.
* Lab 18. Бинарные файлы. Операции над файлами.
* Lab 19. Простые односвязанные списки. Создание списка.
* Lab 20. Метод разделяй и властвуй.

**Лабораторная работа №1.**

**Задание 1.**

**Условие:** ****

**Решение:**

#include <stdio.h>

#include "color.h"

#include <time.h>

#include <stdbool.h>

//14 var

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

const int n **=** 50**;**

int a**[**n**],**i**,**max**,**min**=**0**,**sum**;**

ClearConsoleToColors**(**2**,**7**);**

printf**(**"Массив А:\n"**);**

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

printf**(**"%4i"**,**i**);**

**}**

SetColor**(**5**);**

printf**(**"\n"**);**

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

a**[**i**]** **=** rand**()%**100**-**50**;**

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

**}**

SetColor**(**2**);**

bool b **=** **false;**

int i1**,**i2**;**

max **=** **-**999999**;**

**for(**i**=**1**;**i**<**n**-**1**;**i**++){**

**if((**a**[**i**-**1**]%**2 **==** 1**)&&(**a**[**i**+**1**]%**2 **==** 1**)&&(**a**[**i**-**1**]%**3 **==** 0**)&&(**a**[**i**+**1**]%**3 **==** 0**)){**

sum **=** a**[**i**-**1**]+**a**[**i**+**1**];**

**if(**max**<**sum**)** max **=** sum**;**

b **=** **true;**

i1 **=** i**-**1**;**

i2 **=** i**+**1**;**

**}**

**}**

**if(**b**){**

printf**(**"\nmax[%i,%i](суммы нечетных сосед-х делящихся на 3 элементов) = %i"**,**i1**,**i2**,**max**);**

**}**

//min(отрицательных)

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

**if((**a**[**i**]<**0**)&&(**a**[**i**]<**min**))**

min **=** a**[**i**];**

**}**

printf**(**"\nmin(отрицательных) = %i\n"**,**min**);**

//Если это максимальное значение является последним четным элементом на [n,m],

//найти среднее арифметическое элементов принадлежащих [n,m], иначе – найти минимальное из всех четных.

int n1**,**m**;**

float sred**,**j**;**

printf**(**"\nВведите отрезок [n,m]:\n"**);**

printf**(**"n = "**);** scanf**(**"%i"**,&**n1**);**

printf**(**"m = "**);** scanf**(**"%i"**,&**m**);**

int k**;**

**if(**m**%**2 **==** 0**){**

k **=** m**;**

**}**

**else** k **=** m**-**1**;**

j **=** sum **=** 0**;**

**if(**max **==** k**){**

**for(**i**=**n1**;**i**<=**m**;**i**++){**

sum**+=**i**;**

j**++;**

**}**

sred **=** sum**/**j**;**

printf**(**"Среднее арифметическое элементов принадлежащих [n,m] = %g"**,**sred**);**

**}**

**else{**

int min1**=**999999**;**

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

**if((**a**[**i**]%**2 **==** 0**)&&(**min1**>**a**[**i**])){**

min1 **=** a**[**i**];**

**}**

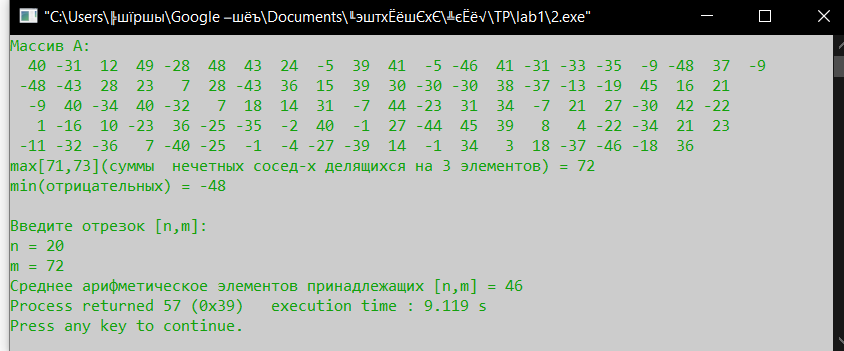
**}**

printf**(**"\nМинимальное из всех четных = %i"**,**min1**);**

**}**

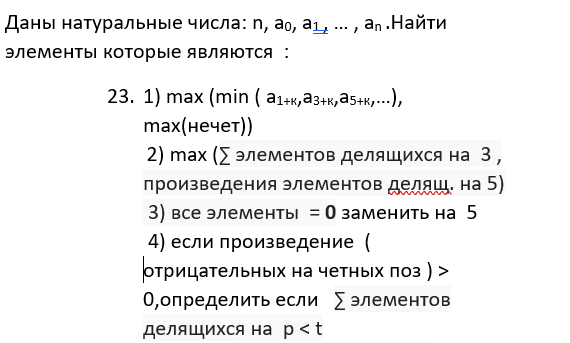
**}**

**Консоль:**

****

**Лабораторная работа №2.**

**Задание 1.**

**Условие:** **Программа:**

#include <stdio.h>

#include "color.h"

#include <time.h>

#include <stdbool.h>

//23 var

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

const int n **=** 20**;**

int a**[**n**],**i**,**max**=-**999999**,**min**=**999999**,**sum **=** 0**,**pr**=**1**;**

ClearConsoleToColors**(**2**,**7**);**

printf**(**"Массив А:\n"**);**

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

printf**(**"%4i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**5**);**

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

a**[**i**]** **=** **(**rand**()%**50**)-**25**;**

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

**}**

SetColor**(**2**);**

//1)max(min(a[1+k,3+k...],max(nechet))

int k**;**

printf**(**"\nK = "**);**scanf**(**"%i"**,&**k**);**

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

**if((**2**\***i**+**1**+**k**)>=**n**)** **break;**

**else**

**if(**a**[**2**\***i**+**1**+**k**]<**min**)** min **=** a**[**2**\***i**+**1**+**k**];**

**}**

**for(**i**=**0**;**i**<**n**/**2**;**i**++){**

**if(**a**[**2**\***i**+**1**]>**max**)** max **=** a**[**2**\***i**+**1**];**

**}**

printf**(**"Max(%i,%i) = %i"**,**min**,**max**,**max**>**min**?**max**:**min**);**

//2)max(sum(el. del. na 3),pr(del. na 5))

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

**if(**a**[**i**]%**3**==**0**)**sum**+=**a**[**i**];**

**if(**a**[**i**]%**5**==**0**&&**a**[**i**]!=**0**)**pr**\*=**a**[**i**];**

**}**

printf**(**"\nMax(%i,%i) = %i"**,**sum**,**pr**,**sum**>**pr**?**sum**:**pr**);**

//3)pervie 3 el, summa indexov kotorih delitsia na 3

**for(**i**=**0**;**i**<**n**-**2**;**i**++){**

**if((**i**+**i**+**1**+**i**+**2**)%**3**==**0**){**

k **=** i**;**

a**[**i**]** **=** a**[**i**+**1**]** **=** a**[**i**+**2**]** **=** 5**;**

**break;**

**}**

**}**

printf**(**"\nЗамена первых 3 элементов, сумма индексов которых делится на 3:\n"**);**

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

printf**(**"%4i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**5**);**

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

**if(**i**==**k**||**i**==**k**+**1**||**i**==**k**+**2**){**

SetColor**(**1**);**

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

SetColor**(**5**);**

**}**

**else** printf**(**"%4i"**,**a**[**i**]);**

**}**

SetColor**(**2**);**

//4)Если произведение(4-х отрицательных от конца на четных позициях)>0, определить если сумма эл-ов делящихся на p<t

pr **=** 1**;**

sum **=** 0**;**

k**=**0**;**

int p**,**t**;**

printf**(**"\np = "**);**scanf**(**"%i"**,&**p**);**

printf**(**"t = "**);**scanf**(**"%i"**,&**t**);**

**for(**i**=**n**-**1**;**i**>=**0**;**i**--){**

**if(**i**%**2**==**0**&&**a**[**i**]<**0**){**

pr**\*=**a**[**i**];**

k**++;**

**}**

**}**

**if(**pr**>**0**){**

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

**if(**a**[**i**]%**p**==**0**)**sum**+=**a**[**i**];**

**}**

**if(**sum**<**t**){**

printf**(**"\nСумма элементов, делящихся на %i (%i) меньше %i"**,**p**,**sum**,**t**);**

**}**

**else{**

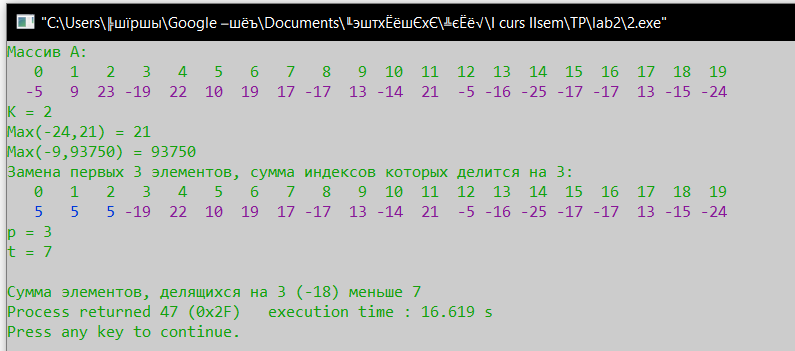
printf**(**"\nСумма элементов, делящихся на %i (%i) не меньше %i"**,**p**,**sum**,**t**);**

**}**

**}**

**}**

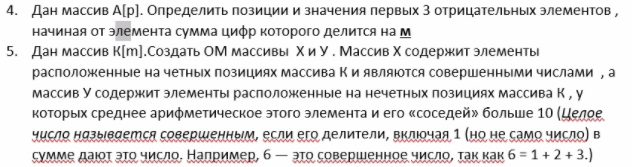
**Консоль:**

****

**Лабораторная работа №3.**

**Задание 1.**

**Условие:**



**Программа:**

#include <stdio.h>

#include "color.h"

#include <time.h>

#include <stdbool.h>

#include <conio.h>

int sum\_num**(**x**){**

int sum**=**0**;**

**if(**x**<**0**){**

x**\*=-**1**;**

**}**

**while(**x**/**10**>**0**){**

sum**+=**x**%**10**;**

x**/=**10**;**

**}**

sum**+=**x**;**

**return** sum**;**

**}**

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

ClearConsoleToColors**(**2**,**7**);**

const int p **=** 50**;**

int l**;**

int a**[**p**],**i**,**m**,**m1**,**k**=**0**;**

int k1**[**p**],**x**[**p**],**y**[**p**],**sov**[**4**]** **=** **{**6**,**28**,**496**,**8128**},**sk**,**sk1**;**

Menu**:**

printf**(**"\nВыберите задание (4 или 5)-->"**);**

scanf**(**"%i"**,&**l**);**

**switch** **(**l**)**

**{**

**case** 4**:**

printf**(**"Массив А:\n"**);**

**for(**i **=** 0**;**i**<**p**;**i**++){**

printf**(**"%4i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**5**);**

**for(**i **=** 0**;**i**<**p**;**i**++){**

a**[**i**]** **=** **(**rand**()%**50**)-**25**;**

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

**}**

SetColor**(**2**);**

printf**(**"\nm = "**);** scanf**(**"%i"**,&**m**);**

printf**(**"\n"**);**

**for(**i**=**0**;**i**<**p**;**i**++){**

printf**(**"%4i"**,**sum\_num**(**a**[**i**]));**

**}**

printf**(**"\n"**);**

**for(**i**=**0**;**i**<**p**;**i**++){**

**if((**sum\_num**(**a**[**i**])%**m**)==**0**){**

m1 **=** i**;**

**break;**

**}**

**}**

k**=**0**;**

printf**(**"\nВыполнение условия задания:\n"**);**

**for(**i **=** 0**;**i**<**p**;**i**++){**

**if(**i**>=**m1**&&**k**<**3**&&**a**[**i**]<**0**){**

k**++;**

SetColor**(**3**);**

printf**(**"%4i"**,**i**);**

SetColor**(**2**);**

**}**

**else{**

printf**(**"%4i"**,**i**);**

**}**

**}**

SetColor**(**5**);**

printf**(**"\n"**);**

k**=**0**;**

**for(**i **=** 0**;**i**<**p**;**i**++){**

**if(**i**>=**m1**&&**k**<**3**&&**a**[**i**]<**0**){**

k**++;**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else{**

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

**}**

**}**

SetColor**(**2**);**

printf**(**"\nПовторить ?(1 - да;2 - завершить программу)-->"**);**

scanf**(**"%i"**,&**l**);**

**if** **(**l **==** 1**){**

**goto** Menu**;**

**}**

**break;**

**case** 5**:**

printf**(**"Массив K:\n"**);**

**for(**i **=** 0**;**i**<**p**;**i**++){**

printf**(**"%3i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**5**);**

**for(**i **=** 0**;**i**<**p**;**i**++){**

k1**[**i**]** **=** rand**()%**30**;**

printf**(**"%3i"**,**k1**[**i**]);**

**}**

SetColor**(**2**);**

sk**=**0**;**sk1**=**0**;**

x**[**0**]** **=** y**[**0**]** **=** 0**;**

printf**(**"\nМассив К:\n"**);**

**for(**i**=**0**;**i**<**p**;**i**++){**

**if((**i**%**2**==**0**)&&(**k1**[**i**]==**sov**[**0**]||**k1**[**i**]==**sov**[**1**]||**k1**[**i**]==**sov**[**2**]||**k1**[**i**]==**sov**[**3**])){**

SetColor**(**8**);**

printf**(**"%3i"**,**i**);**

SetColor**(**2**);**

**}**

**else** **if((**i**%**2**!=**0**)&&**i**>**0**&&**i**<**p**-**1**&&(((**k1**[**i**-**1**]+**k1**[**i**]+**k1**[**i**+**1**])/**3.0**)>**10**)){**

SetColor**(**9**);**

printf**(**"%3i"**,**i**);**

SetColor**(**2**);**

**}**

**else**

printf**(**"%3i"**,**i**);**

**}**

printf**(**"\n"**);**

**for(**i**=**0**;**i**<**p**;**i**++){**

**if((**i**%**2**==**0**)&&(**k1**[**i**]==**sov**[**0**]||**k1**[**i**]==**sov**[**1**]||**k1**[**i**]==**sov**[**2**]||**k1**[**i**]==**sov**[**3**])){**

x**[**sk**]** **=** k1**[**i**];**

sk**++;**

SetColor**(**8**);**

printf**(**"%3i"**,**k1**[**i**]);**

SetColor**(**2**);**

**}**

**else** **if((**i**%**2**!=**0**)&&**i**>**0**&&**i**<**p**-**1**&&(((**k1**[**i**-**1**]+**k1**[**i**]+**k1**[**i**+**1**])/**3.0**)>**10**)){**

y**[**sk1**]** **=** k1**[**i**];**

sk1**++;**

SetColor**(**9**);**

printf**(**"%3i"**,**k1**[**i**]);**

SetColor**(**2**);**

**}**

**else**

printf**(**"%3i"**,**k1**[**i**]);**

**}**

printf**(**"\nРазделение на массивы X и Y:\n"**);**

**if(**x**[**0**]!=**0**){**

printf**(**"Массив X(расположены на нечетных позициях и являются совершенными числами):\n"**);**

**for(**i **=** 0**;**i**<**sk**;**i**++){**

printf**(**"%3i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**8**);**

**for(**i **=** 0**;**i**<**sk**;**i**++){**

printf**(**"%3i"**,**x**[**i**]);**

**}**

SetColor**(**2**);**

**}**

**else{**

printf**(**"\nНи одно значение не подошло под условие для массива X!!!"**);**

**}**

**if(**y**[**0**]!=**0**){**

printf**(**"\nМассив Y(расположены на нечетных позициях и у которых среднее арифметическое этого элемента и его соседей больше 10):\n"**);**

**for(**i **=** 0**;**i**<**sk1**;**i**++){**

printf**(**"%3i"**,**i**);**

**}**

printf**(**"\n"**);**

SetColor**(**9**);**

**for(**i **=** 0**;**i**<**sk1**;**i**++){**

printf**(**"%3i"**,**y**[**i**]);**

**}**

SetColor**(**2**);**

**}**

**else{**

printf**(**"\nНи одно значение не подошло под условие для массива Y!!!"**);**

**}**

printf**(**"\nПовторить ?(1 - да;2 - завершить программу)-->"**);**

scanf**(**"%i"**,&**l**);**

**if** **(**l **==** 1**){**

**goto** Menu**;**

**}**

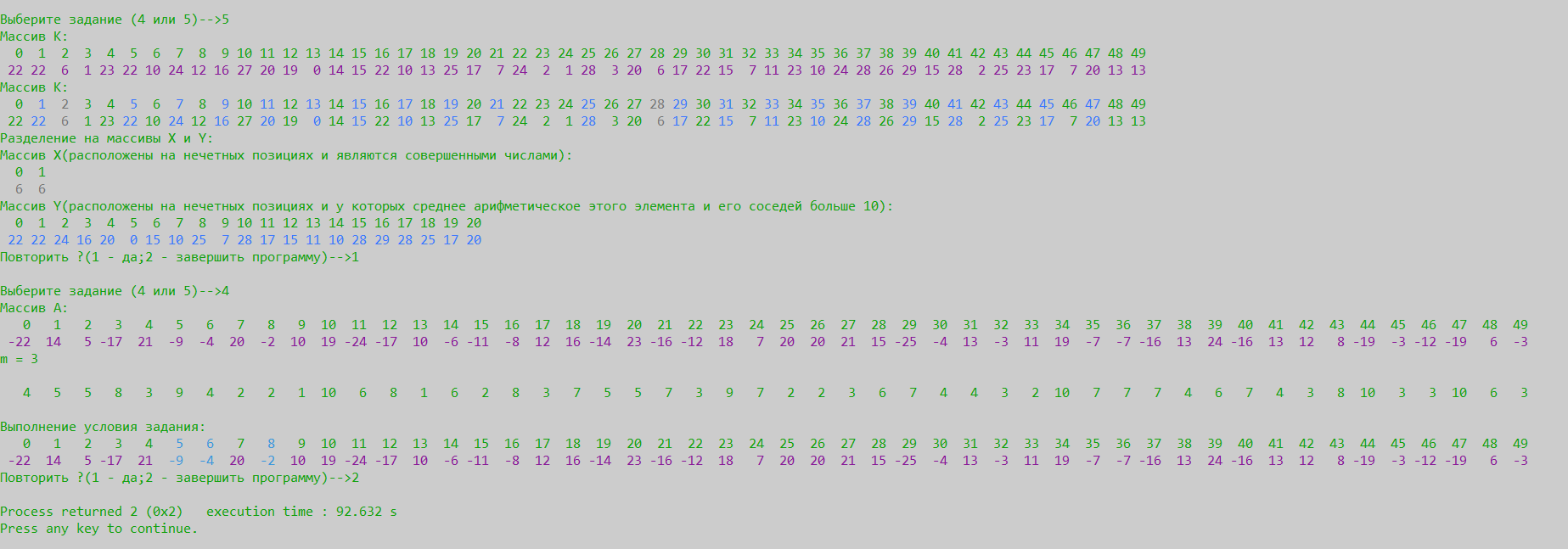
**break;**

**default:** printf**(**"\nНеправильное значение, попробуйте еще раз!!"**);** **goto** Menu**;**

**}**

**}**

**Консоль:**

****

**Лабораторная работа №4.**

**Задание 1.**

**Условие:**

**Программа:**

#include <stdio.h>

#include <math.h>

#include <color.h>

#include <stdbool.h>

#include <conio.h>

//16,685,684

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

const int n **=** 4**;**

float d**[**n**][**n**],**a**[**n**][**n**],**sum**,**sr**,**s**=**0**;**

int i**,**j**,**k**=**0**;**

printf**(**"Массив D:\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

d**[**i**][**j**]** **=** abs**(**exp**(**i**)-**exp**(**j**))** **+** **(**i**+**j**)/**2.0 **+**1**;**

printf**(**"%5.1f"**,**d**[**i**][**j**]);**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

printf**(**"\nВывод ответа. Массив A(среднее арифметическое индексов соседей если сумма соседей больше):\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**i**==**0**){**

**if(**j**==**0**){**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**][**j**+**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**+**j**+**1**;**

a**[**i**][**j**]** **=** sr**/**4**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else** **if(**j**==**n**-**1**){**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**4**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else{**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**][**j**+**1**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**+**j**+**1**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**6**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**}**

**else** **if(**i**==**n**-**1**){**

**if(**j**==**0**){**

sum **=** d**[**i**-**1**][**j**]+**d**[**i**][**j**+**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**-**1**+**j**+**i**+**j**+**1**;**

a**[**i**][**j**]** **=** sr**/**4**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else** **if(**j**==**n**-**1**){**

sum **=** d**[**i**-**1**][**j**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**-**1**+**j**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**4**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else{**

sum **=** d**[**i**-**1**][**j**]+**d**[**i**][**j**+**1**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**-**1**+**j**+**i**+**j**+**1**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**6**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**}**

**else{**

**if(**j**==**0**){**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**-**1**][**j**]+**d**[**i**][**j**+**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**-**1**+**j**+**i**+**j**+**1**;**

a**[**i**][**j**]** **=** sr**/**6**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else** **if(**j**==**n**-**1**){**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**-**1**][**j**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**-**1**+**j**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**6**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**else{**

sum **=** d**[**i**+**1**][**j**]+**d**[**i**-**1**][**j**]+**d**[**i**][**j**+**1**]+**d**[**i**][**j**-**1**];**

**if(**d**[**i**][**j**]<**sum**){**

sr **=** i**+**1**+**j**+**i**-**1**+**j**+**i**+**j**+**1**+**i**+**j**-**1**;**

a**[**i**][**j**]** **=** sr**/**8**;**

s**+=**a**[**i**][**j**];** k**++;**

SetColor**(**3**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**else** **{**

a**[**i**][**j**]** **=** d**[**i**][**j**];**

SetColor**(**5**);**

printf**(**"%5.1f"**,**a**[**i**][**j**]);**

**}**

**}**

**}**

**}**

printf**(**"\n"**);**

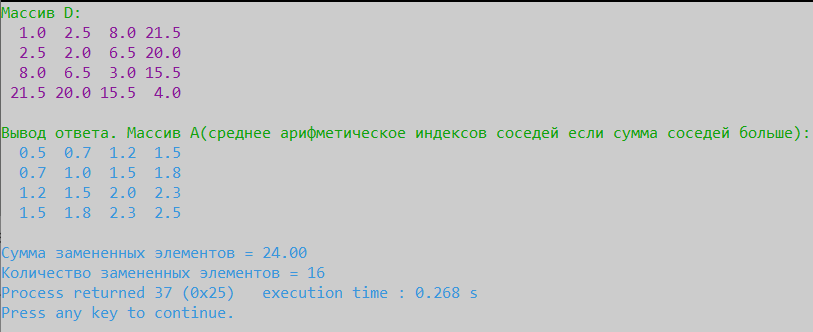
**}**

printf**(**"\nСумма замененных элементов = %.2f"**,**s**);**

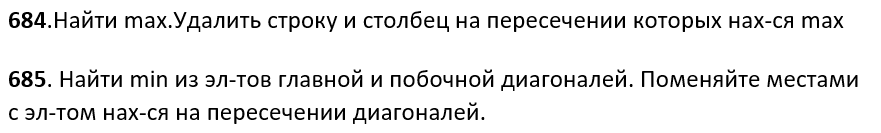
printf**(**"\nКоличество замененных элементов = %i"**,**k**);**

**}**

**Консоль:**

****

**Задание 2.**

**Условие:**

**Программа:**

#include <stdio.h>

#include <math.h>

#include <color.h>

#include <stdbool.h>

#include <conio.h>

#include <time.h>

//16,685,684

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

const int n **=** 5**;**

int i**,**j**,**k**=**0**,**a**[**n**][**n**],**c**[**n**][**n**],**max**=-**99999**,**min**=**99999**,**imax**,**jmax**,**imin**,**jmin**;**

bool bi**=**false**,**bj**=**false**;**

printf**(**"Массив A:\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

a**[**i**][**j**]** **=** c**[**i**][**j**]** **=** rand**()%**100**;**

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

**if(**a**[**i**][**j**]>**max**){**

max **=** a**[**i**][**j**];**

imax **=** i**;**

jmax **=** j**;**

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

printf**(**"(684)Массив B(с удаленными строками и столбцами на max):\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**i**==**imax**&&**j**==**jmax**){**

SetColor**(**3**);**

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

**}**

**else** **if(**i**==**imax**||**j**==**jmax**){**

SetColor**(**6**);**

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

**}**

**else{**

SetColor**(**5**);**

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

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**5**);**

printf**(**"\n\n"**);**

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

**if(**i**==**imax**){**

**continue;**

**}**

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**j**==**jmax**){**

**continue;**

**}**

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

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

printf**(**"\n\n(685)Массив С(пересечение двух диагоналей заменено минимумом элементов главное и побочной диагоналей):\n"**);**

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

**if(**min**>**c**[**i**][**i**]){**

min **=** c**[**i**][**i**];**

imin **=** jmin **=** i**;**

**}**

**if(**min**>**a**[**i**][**n**-**i**-**1**]){**

min **=** c**[**i**][**n**-**i**-**1**];**

imin **=** i**;**

jmin **=** n**-**i**-**1**;**

**}**

**}**

SetColor**(**5**);**

k**=**c**[**n**/**2**][**n**/**2**];**

c**[**n**/**2**][**n**/**2**]** **=** c**[**imin**][**jmin**];**

c**[**imin**][**jmin**]** **=** k**;**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if((**i**==**n**/**2**&&**j**==**n**/**2**)||(**i**==**imin**&&**j**==**jmin**)){**

SetColor**(**6**);**

printf**(**"%4i"**,**c**[**i**][**j**]);**

SetColor**(**5**);**

**}**

**else{**

printf**(**"%4i"**,**c**[**i**][**j**]);**

**}**

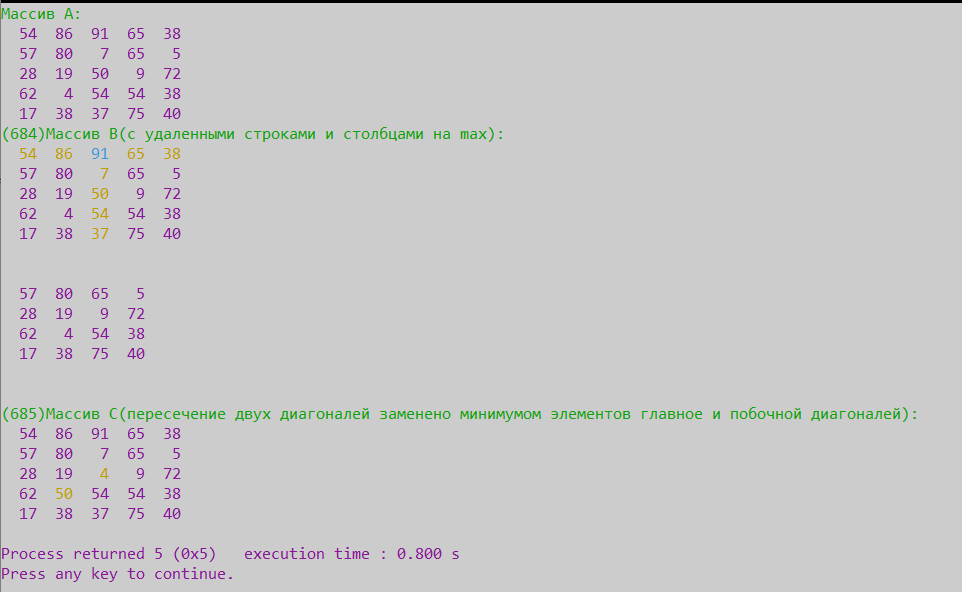
**}**

printf**(**"\n"**);**

**}**

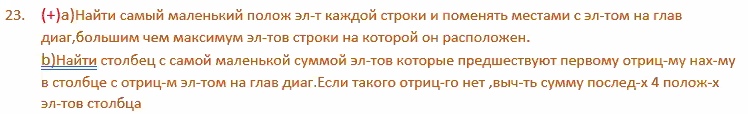
**}**

**Консоль:**

****

**Лабораторная работа №5.**

**Задание 1.**

**Условие:**

**Программа:**

#include <stdio.h>

#include <math.h>

#include <color.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

//23

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

int k**,**n**;**

char ch**;**

printf**(**"Введите размерность матрицы N = "**);**

scanf**(**"%i"**,&**n**);**

int a**[**n**][**n**],**i**,**j**;**

bool ba**,**bb**;**

M1**:**

printf**(**"\nВыберите случай:\na)Найти самый маленький полож эл-т каждой строки и поменять местами с эл-ом на глав диаг, большим чем максимум эл-в строки на которой он расположен\nb)Найти столбец с самой маленькой суммой эл-ов, которые предшествуют первому отриц-му нах-му в столбце с отриц-м эл-ом на клавной диаг. Если такого отриц-го нет, выч-ть сумму последних 4 полож-х эл-ов столбца\n-->"**);**

ch **=** getche**();**

ba**=**bb**=**false**;**

**switch(**ch**){**

**case** 'a'**:**

ba**=**true**;**

**break;**

**case** 'b'**:**

bb**=**true**;**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

M2**:**

printf**(**"\n1)Массив вводим с клавиатуры.\n2)Массив генерируем случайно\n-->"**);**

ch **=** getche**();**

printf**(**"\n"**);**

**switch(**ch**){**

**case** '1'**:**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

printf**(**"A[%i,%i] = "**,**i**,**j**);**

scanf**(**"%i"**,&**a**[**i**][**j**]);**

**}**

**}**

**break;**

**case** '2'**:**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

a**[**i**][**j**]** **=** **(**rand**()%**100**)-**50**;**

**}**

**}**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M2**;**

**}**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

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

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

int min**=**99999**,**max**=-**99999**,**c**,**j1**=-**1**,**nn**=**n**/**2**;**

bool neg**=**false**;**

int sum**[**n**],**sum\_el**=**0**,**sum\_st**=**99999**,**i1**;**

**if(**ba**){**

printf**(**"Выполнение условия a):\n"**);**

SetColor**(**5**);**

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

min **=** 99999**;**

**for(**k**=**0**;**k**<**n**;**k**++){**

**if((**i**==**nn**)&&(**max**<**a**[**i**][**k**])){**

max **=** a**[**i**][**k**];**

j1 **=** k**;**

**}**

**if((**min**>**a**[**i**][**k**])&&(**a**[**i**][**k**]>**0**)){**

min **=** a**[**i**][**k**];**

**}**

**}**

**if(**i**==**nn**){**

c **=** a**[**nn**][**nn**];**

a**[**nn**][**nn**]** **=** max**;**

a**[**nn**][**j1**]** **=** c**;**

**}**

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**a**[**i**][**j**]==**min**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else** **if((**i**==**nn**&&**j**==**nn**)||(**i**==**nn**&&**j**==**j1**)){**

SetColor**(**4**);**

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

SetColor**(**5**);**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

**}**

**else{**

printf**(**"Выполнение условия b):\n"**);**

SetColor**(**5**);**

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

sum**[**i**]=**0**;**

**}**

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

neg**=**false**;**

**if(**a**[**i**][**i**]<**0**){**

neg **=** true**;**

**}**

**if(**neg**){**

sum\_el **=** 0**;**

**for(**j**=**0**;**j**<**i**;**j**++){**

**if(**a**[**j**][**i**]>**0**){**

sum\_el**+=**a**[**j**][**i**];**

**}**

**else** **break;**

**}**

**if(**sum\_st**>**sum\_el**){**

sum\_st **=** sum\_el**;**

i1**=**i**;**

**}**

**}**

**else{**

**for(**j**=**n**-**1**,**k**=**0**;**j**>=**0**,**k**<**5**;**k**++,**j**--){**

sum**[**i**]+=**a**[**j**][**i**];**

**}**

**}**

**}**

printf**(**"\n"**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**j**==**i1**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

printf**(**"\n\n"**);**

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

**if(**sum**[**i**]!=**0**){**

printf**(**"%4i"**,**sum**[**i**]);**

**}**

**else{**

printf**(**" "**);**

**}**

**}**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** M1**;**

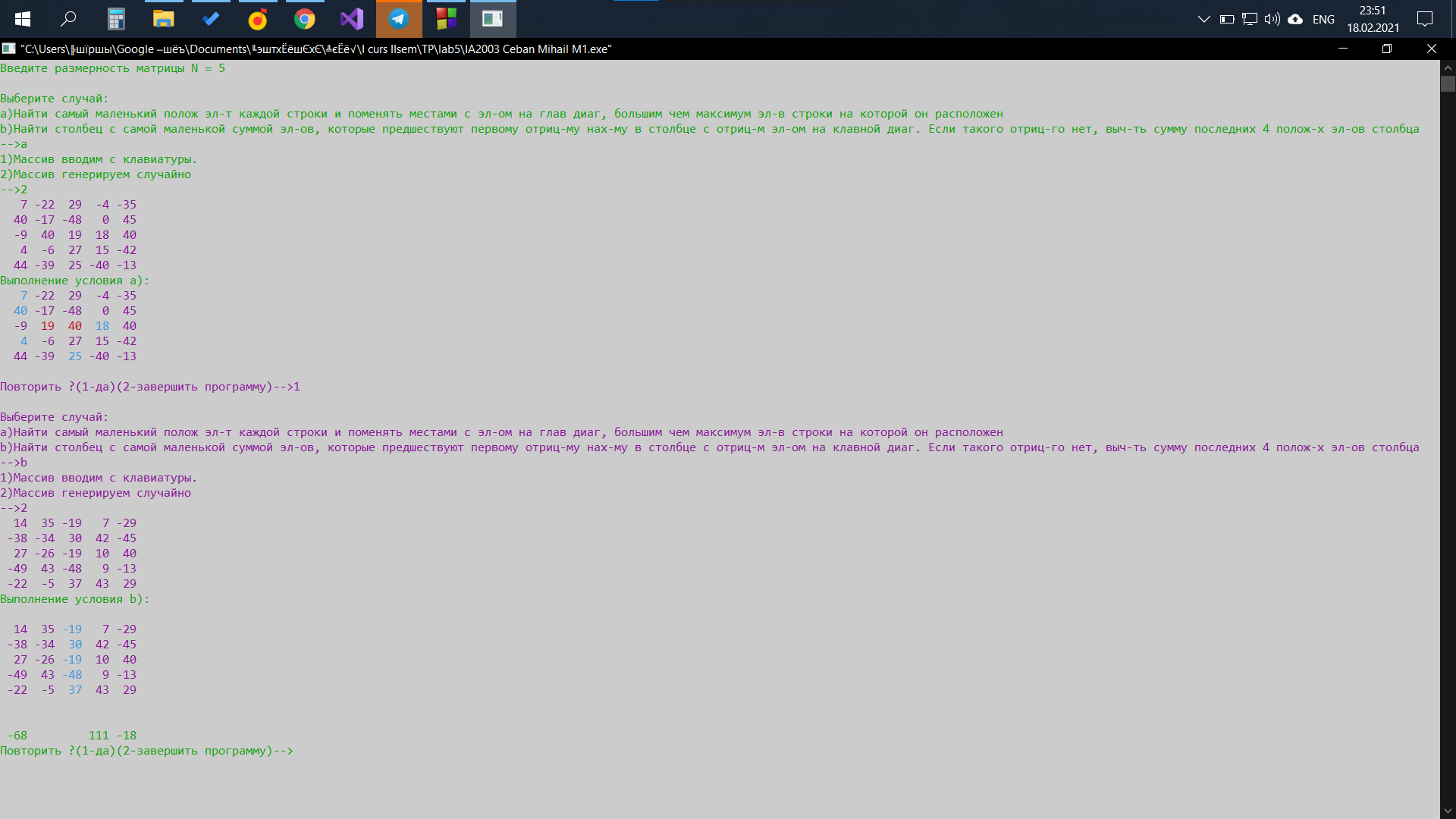
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

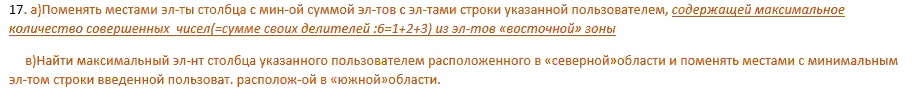
**Консоль:**

****

**Лабораторная работа №6.**

**Задание 1.**

**Условие:**

****

**Программа:**

#include <stdio.h>

#include <math.h>

#include <color.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

//17

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

int k**,**n**;**

char ch**;**

printf**(**"Введите размерность матрицы N = "**);**

scanf**(**"%i"**,&**n**);**

int a**[**n**][**n**],**i**,**j**,**sov**[**4**]={**6**,**28**,**496**,**8128**};**

int nStroki**,**sum**,**min**=**99999**,**max**=-**99999**,**countK**,**iFix**=**1**,**iMaxSov**=**0**,**jMinStolb**,**vrem**;**

int j1**,**i1**,**iMaxSev**,**jMinSouth**;**

bool ba**,**bb**;**

M1**:**

printf**(**"\nВыберите случай:\na)Поменять местами эл-ты столбца с мин-ой суммой эл-ов с эл-тами строки, указанной пользователем, содержащей максимальное количество совершенных чисел(=сумме своих делителей :6=1+2+3) из элементов восточной зоны.\nb)Найти максимальный элемент столбца, указанного пользователем, расположенного в северной области и поменять местами с минимальным элементом строки, введенной пользователем, расположенной в южной области\n-->"**);**

ch **=** getche**();**

ba**=**bb**=**false**;**

**switch(**ch**){**

**case** 'a'**:**

ba**=**true**;**

**break;**

**case** 'b'**:**

bb**=**true**;**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

printf**(**"\nМатрица:\n"**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

a**[**i**][**j**]** **=** **(**rand**()%**7**);**

**}**

**}**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

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

**}**

printf**(**"\n"**);**

**}**

SetColor**(**2**);**

**if(**ba**){**

printf**(**"Матрица с восточной зоной :\n"**);**

SetColor**(**5**);**

countK **=** 0**;**

max **=** **-**99999**;**

min **=** 99999**;**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

sum **=** 0**;**

**if(**i**==**j**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**+**j **==** n**-**1**){**

SetColor**(**4**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**<**j**&&**i**+**j**>**n**-**1**){**

SetColor**(**6**);**

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

SetColor**(**5**);**

**if(**i**==**iFix**){**

**for(**k**=**0**;**k**<**4**;**k**++){**

**if(**a**[**i**][**j**]==**sov**[**k**]){**

countK**++;**

**break;**

**}**

**}**

**}**

**else{**

**if(**max**<**countK**){**

max **=** countK**;**

iMaxSov **=** iFix**;**

**}**

countK **=** 0**;**

iFix **=** i**;**

**for(**k**=**0**;**k**<**4**;**k**++){**

**if(**a**[**i**][**j**]==**sov**[**k**]){**

countK**++;**

**break;**

**}**

**}**

**}**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

**if(**iMaxSov**==**0**){**

printf**(**"\nМатрица не содержит совершенных чисел в восточной зоне!Создайте новую!"**);**

**goto** M1**;**

**}**

M2**:**

printf**(**"\nВведите номер строки матрицы (0-%i):"**,**n**-**1**);**

scanf**(**"%i"**,&**nStroki**);**

**if(**nStroki**!=**iMaxSov**){**

printf**(**"\nЭта строка не содержит максимальное количество совершенных чисел из восточной зоны!\nСоветую ввести номер строки %i"**,**iMaxSov**);**

**goto** M2**;**

**}**

//нахождение столбца с минимальной суммой элементов

**for(**j**=**0**;**j**<**n**;**j**++){**

sum **=** 0**;**

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

sum **+=** a**[**i**][**j**];**

**}**

**if(**sum**<**min**){**

min **=** sum**;**

jMinStolb **=** j**;**

**}**

**}**

//zamena

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

vrem **=** a**[**iMaxSov**][**i**];**

a**[**iMaxSov**][**i**]** **=** a**[**i**][**jMinStolb**];**

a**[**i**][**jMinStolb**]** **=** vrem**;**

//a[iMaxSov][i] += a[i][jMinStolb];

//a[i][jMinStolb] = a[iMaxSov][i] - a[i][jMinStolb];

//a[iMaxSov][i] -= a[i][jMinStolb];

**}**

//vivod

printf**(**"\nВывод ответа:\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**j**==**jMinStolb**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**==**iMaxSov**){**

SetColor**(**4**);**

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

SetColor**(**5**);**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

**}**

**else{**

printf**(**"\nВведите номер столбца матрицы(0-%i):"**,**n**-**1**);**

scanf**(**"%i"**,&**j1**);**

printf**(**"\nВведите номер строки матрицы(0-%i):"**,**n**-**1**);**

scanf**(**"%i"**,&**i1**);**

printf**(**"Матрица с северной и южной зонами :\n"**);**

SetColor**(**5**);**

countK **=** 0**;**

max **=** **-**99999**;**

min **=** 99999**;**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

sum **=** 0**;**

**if(**i**==**j**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**+**j **==** n**-**1**){**

SetColor**(**4**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**<**j**&&**i**+**j**<**n**-**1**){**

SetColor**(**6**);**

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

SetColor**(**5**);**

**if(**j**==**j1**&&**a**[**i**][**j**]>**max**){**

max **=** a**[**i**][**j**];**

iMaxSev **=** i**;**

**}**

**}**

**else** **if(**i**>**j**&&**i**+**j**>**n**-**1**){**

SetColor**(**8**);**

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

SetColor**(**5**);**

**if(**i**==**i1**&&**min**>**a**[**i**][**j**]){**

min **=** a**[**i**][**j**];**

jMinSouth **=** j**;**

**}**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

vrem **=** a**[**iMaxSev**][**j1**];**

a**[**iMaxSev**][**j1**]** **=** a**[**i1**][**jMinSouth**];**

a**[**i1**][**jMinSouth**]** **=** vrem**;**

SetColor**(**2**);**

printf**(**"\nВывод ответа:\n"**);**

SetColor**(**5**);**

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

**for(**j**=**0**;**j**<**n**;**j**++){**

**if(**i**==**i1**&&**j**==**jMinSouth**){**

SetColor**(**3**);**

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

SetColor**(**5**);**

**}**

**else** **if(**i**==**iMaxSev**&&**j**==**j1**){**

SetColor**(**4**);**

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

SetColor**(**5**);**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** M1**;**

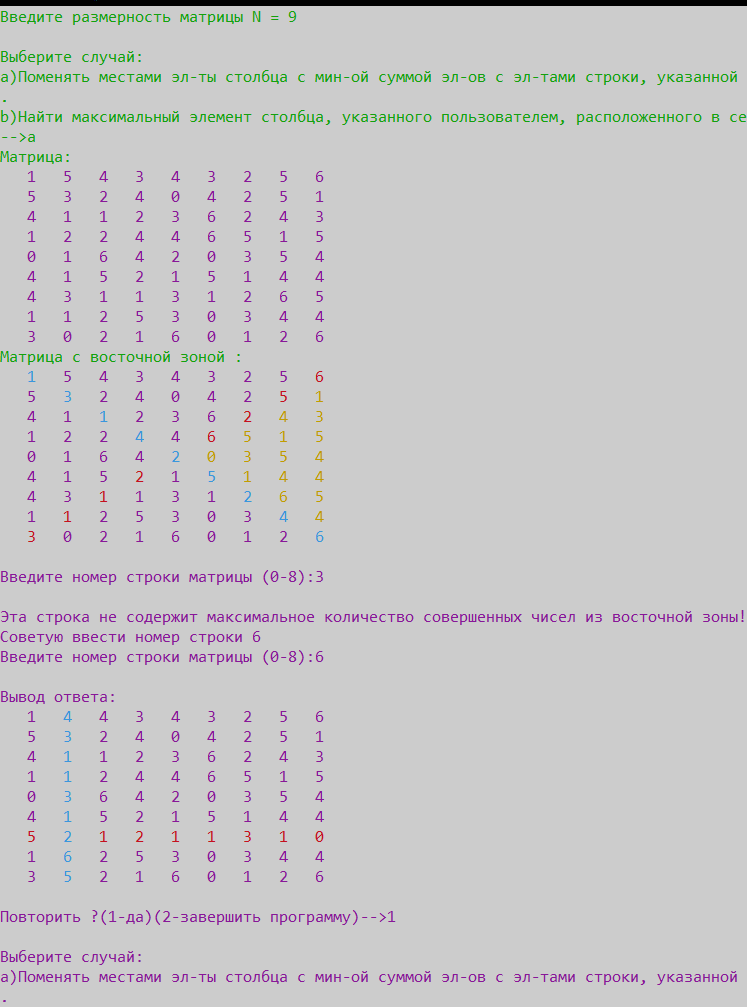
**case** 2**:break;**

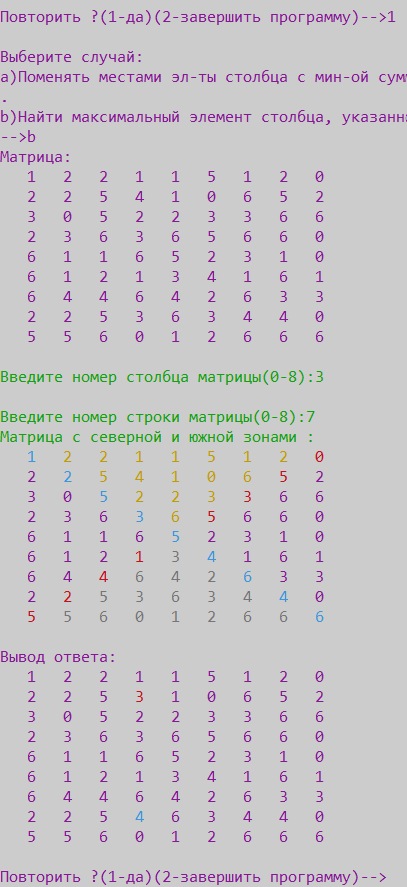
**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

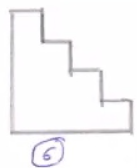
****

**Лабораторная работа №7.**

**Задание 1.**

**Условие:**

**Используя функции построить**



**Программа:**

#include <stdio.h>

#include <color.h>

#include <gotoxy.h>

//6

void f1**(){**

printf**(**"\_\_"**);**

printf**(**"\n"**);**

**}**

void f2**(**n**){**

int i**,**j**;**

**for(**i**=**0**;**i**<**2**;**i**++){**

printf**(**"|"**);**

gotoxy**(**3**\***n**,**2**\***n**-**1**+**i**);**

printf**(**"|"**);**

printf**(**"\n"**);**

**}**

gotoxy**(**3**\***n**+**1**,**2**\***n**-**1**+**1**);**

**}**

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

printf**(**" "**);**

int i**;**

**for(**i**=**1**;**i**<**5**;**i**++){**

f1**();**

f2**(**i**);**

**}**

printf**(**"\n"**);**

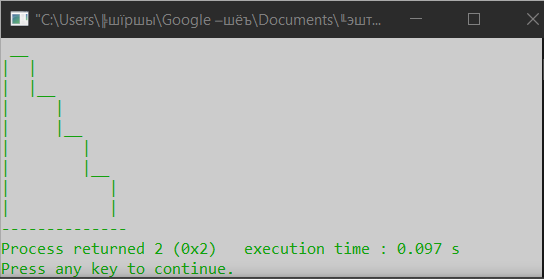
**for(**i**=**0**;**i**<**7**;**i**++){**

printf**(**"--"**);**

**}**

**}**

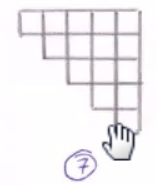
**Консоль:**

****

**Задание 2.**

**Условие:**

**Используя функции построить**



**Программа:**

#include <stdio.h>

#include <math.h>

#include <color.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

//17

void f1**(**n**){**

int i**;**

f3**(**n**);**

**for(**i**=**0**;**i**<**6**-**n**;**i**++){**

printf**(**"\_\_\_"**);**

**}**

printf**(**"\_"**);**

printf**(**"\n"**);**

**}**

void f2**(){**

int i**,**j**;**

**for(**j**=**5**;**j**>**0**;**j**--){**

f1**(**6**-**j**);**

f3**(**6**-**j**);**

**for(**i**=**j**+**1**;**i**>**0**;**i**--){**

printf**(**"| "**);**

**if(**i**==**1**){**

printf**(**"\n"**);**

**}**

**}**

f4**(**6**-**j**);**

**}**

**}**

void f3**(**n**){**

int i**;**

**for(**i**=**n**;**i**>**0**;**i**--){**

printf**(**" "**);**

**}**

**}**

void f4**(**n**){**

int i**;**

f3**(**n**);**

**for(**i**=**0**;**i**<**6**-**n**;**i**++){**

printf**(**"---"**);**

**}**

printf**(**"-"**);**

printf**(**"\n"**);**

**}**

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

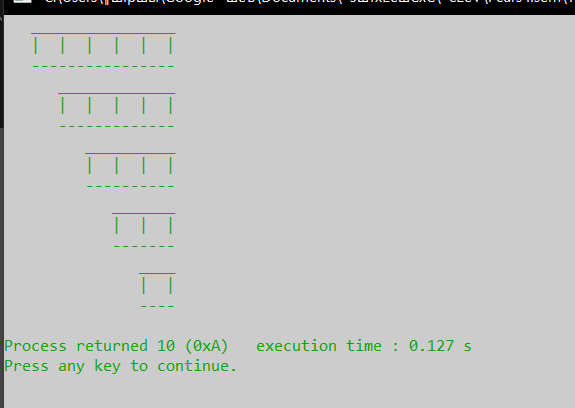
ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

f2**();**

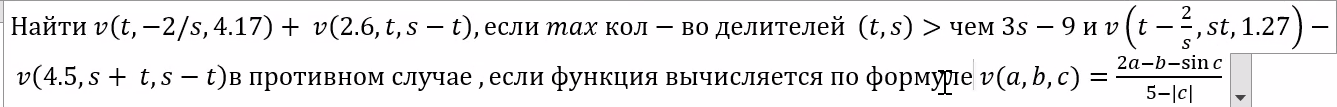
**}**

**Консоль:**

****

**Задание 3.**

**Условие:**



**Программа:**

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

//22

float f2**(**float e**);**

bool isValid2**(**float j**){**

**if(**j**<=**0**){**

printf**(**"\nb должно быть больше 0 по условию логарифма"**);**

**return** false**;**

**}**

**else** **return** true**;**

**}**

bool isValid1**(**float p**,** float q**){**

**if((**6**\***p**-**f2**(**q**))==**0**){**

printf**(**"\nДеление на ноль невозможно!(6\*x-f2(%g)=0)"**,**q**);**

**return** false**;**

**}**

**else** **return** true**;**

**}**

float f1**(**float x**,**float y**,**float z**){**

**return** **(**8**\***x**-**log**(**y**))/(**6**\***x**-**f2**(**z**));**

**}**

float f2**(**float e**){**

**return** **(**7**-**5**\***abs**(**e**));**

**}**

main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

ClearConsoleToColors**(**2**,**7**);**

srand**(**time**(NULL));**

float a**,**b**,**c**;**

int k**;**

M1**:** printf**(**"Введите числа a,b,c:\n"**);**

printf**(**"a = "**);** scanf**(**"%f"**,&**a**);**

printf**(**"b = "**);** scanf**(**"%f"**,&**b**);**

printf**(**"c = "**);** scanf**(**"%f"**,&**c**);**

**if(**isValid2**(**b**)&&**isValid1**(**a**,**c**)){**

printf**(**"\nF2 = %g"**,**f2**(**c**));**

printf**(**"\nF1 = %g"**,**f1**(**a**,**b**,**c**));**

**}**

**else{**

printf**(**"\nПопробуйте снова:\n"**);**

**goto** M1**;**

**}**

M2**:**

printf**(**"\nХотите попробовать снова?(1-да|2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:** **goto** M1**;** **break;**

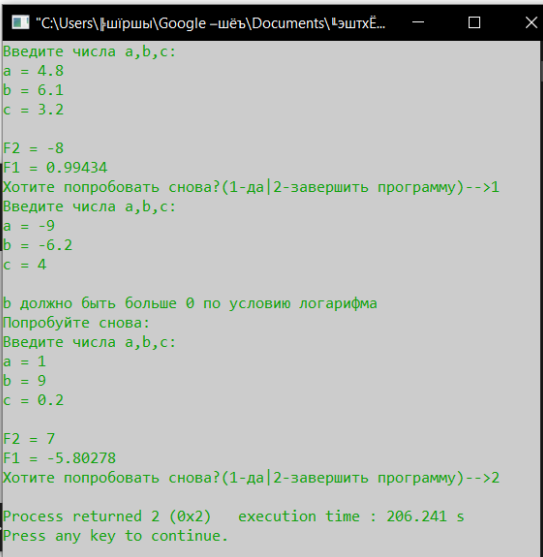
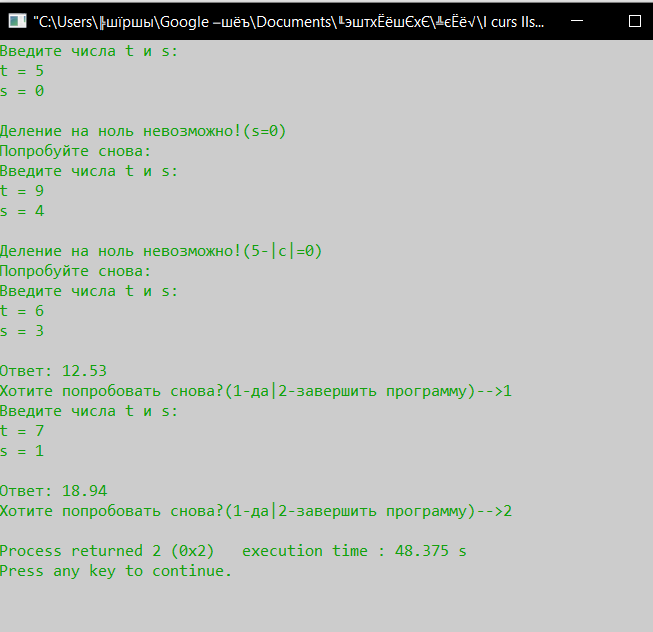
**case** 2**:** **break;**

**default:**printf**(**"Недопустимое значение, попробуйте снова!"**);** **goto** M2**;**

**}**

**}**

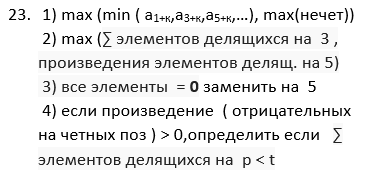
**Консоль:**

****

**Лабораторная работа №8.**

**Задание 1.**

**Условие:**



**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <workWithOMArray.h>

#define forLoop for(int i=0;i < n;i++)

void f1**(**int x**[**20**],**int n**){**

int k**,**min**=**999999**,**max**=-**999999**;**

printf**(**"\nK = "**);**

scanf**(**"%i"**,&**k**);**

forLoop**{**

**if((**2**\***i**+**1**+**k**)>=**n**)** **break;**

**else**

min **=** Min**(**x**[**2**\***i**+**1**+**k**],**min**);**

**}**

forLoop**{**

i**++;**

**if((**2**\***i**+**1**+**k**)>=**n**)** **break;**

**else**

max **=** Max**(**x**[**2**\***i**+**1**],**max**);**

**}**

printf**(**"Max(%i,%i) = %i"**,**min**,**max**,**Max**(**min**,**max**));**

**}**

void f2**(**int x**[**20**],**int n**){**

int sum**=**0**,**pr**=**1**;**

forLoop**{**

**if(**del**(**x**[**i**],**3**))**sum**+=**x**[**i**];**

**if(**del**(**x**[**i**],**5**)&&**x**[**i**]!=**0**)**pr**\*=**x**[**i**];**

**}**

printf**(**"\nMax(%i,%i) = %i"**,**sum**,**pr**,**Max**(**sum**,**pr**));**

**}**

void f3**(**int x**[**20**],**int n**){**

printf**(**"\nРешение\n"**);**

forLoop**{**

**if(**x**[**i**]==**0**){**

x**[**i**]=**5**;**

**}**

**}**

outputArr**(**x**,**n**);**

**}**

void f4**(**int x**[**20**],**int n**){**

int pr**=**1**,**sum**=**0**,**k**=**0**;**

int p**,**t**;**

printf**(**"\np = "**);**

scanf**(**"%i"**,&**p**);**

printf**(**"t = "**);**

scanf**(**"%i"**,&**t**);**

forLoop**{**

**if(**x**[**i**]<**0**){**

pr**\*=**x**[**i**];**

**}**

i**++;**

**}**

**if(**pr**>**0**){**

forLoop**{**

**if(**del**(**x**[**i**],**p**))**sum**+=**x**[**i**];**

**}**

**if(**sum**<**t**){**

printf**(**"\nСумма элементов, делящихся на %i (%i) меньше %i"**,**p**,**sum**,**t**);**

**}**

**else{**

printf**(**"\nСумма элементов, делящихся на %i (%i) не меньше %i"**,**p**,**sum**,**t**);**

**}**

**}**

**else{**

printf**(**"\nПроизведение отрицательных меньше нуля!!"**);**

**}**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

Menu**:**

int k**,**N**;**

char ch**;**

printf**(**"\nВведите размерность массива N = "**);**

scanf**(**"%i"**,&**N**);**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

int a**[**N**];**

**switch(**k**){**

**case** 1**:**

inputArrKeyB**(**a**,**N**);**

**break;**

**case** 2**:**

inputArrRand**(**a**,**N**);**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

outputArr**(**a**,**N**);**

M1**:**

printf**(**"\nВыберите случай:\n1)max(min(a[1+k],a[3+k],..),max(нечет))\n2)max(summ(эл-ов делящ. на 3), произведения элементов делящихся на 5)\n3)все эл-ты - 0 заменить на 5\n4)если произведение(отрицательных на четных поз)>0, определить если сумма элементов делящихся на p < t\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

f1**(**a**,**N**);**

**break;**

**case** '2'**:**

f2**(**a**,**N**);**

**break;**

**case** '3'**:**

f3**(**a**,**N**);**

**break;**

**case** '4'**:**

f4**(**a**,**N**);**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

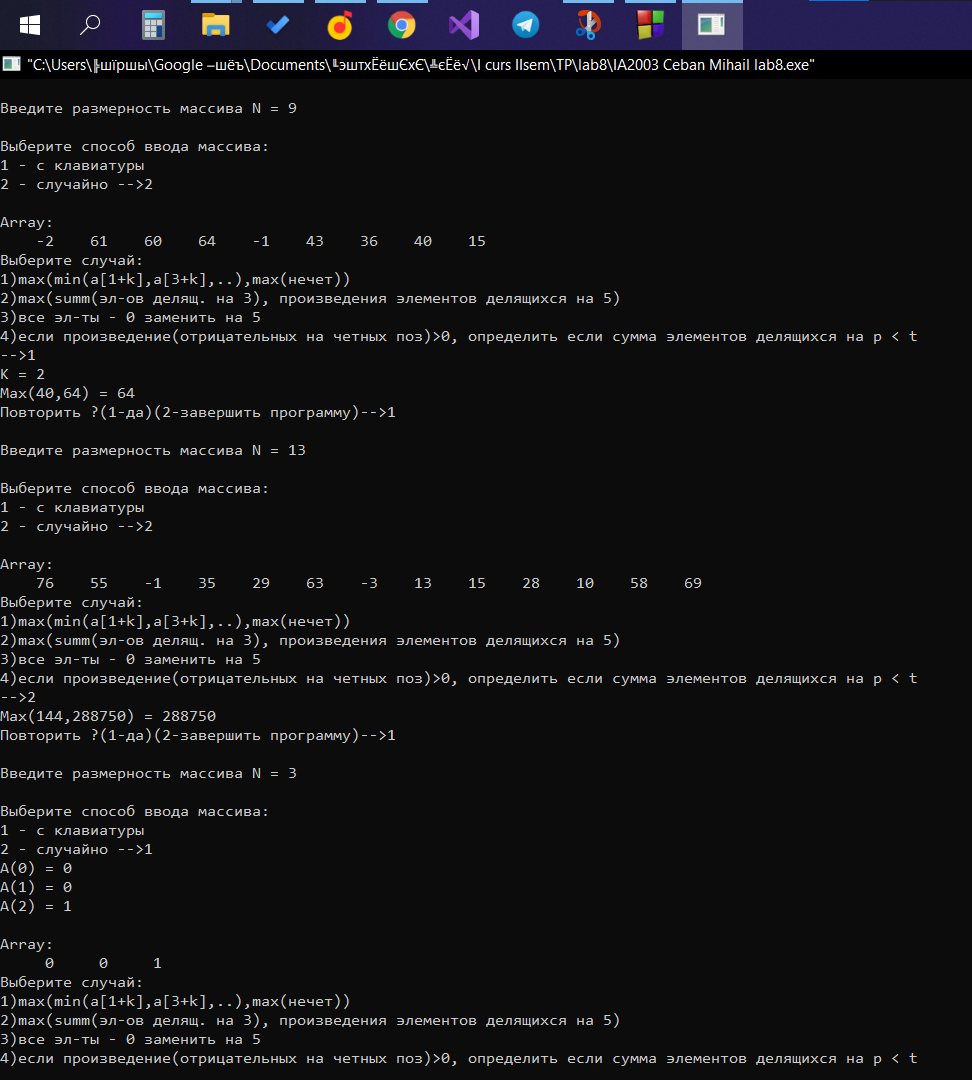
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль**



**Лабораторная работа №9.**

**Задание 1.**

**Условие:**

## 

**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <workWithOMArray.h>

#include <color.h>

#define forI for(int i=0;i < m;i++)

#define forJ for(int j=0;j < m;j++)

bool isExcess**(**int x**){**

int sum**=**0**;**

**if(**x**<**0**){**

x**\*=-**1**;**

**}**

**for(**int i**=**1**;**i**<=**x**/**2**;**i**++){**

**if(**x**%**i**==**0**){**

sum**+=**i**;**

**}**

**}**

**if(**sum**>**x**){**

**return** **true;**

**}**

**else{**

**return** **false;**

**}**

**}**

bool isSov**(**int x**){**

int sov**[**4**]={**6**,**28**,**496**,**8128**};**

bool b**=false;**

**for(**int i**=**0**;**i**<**4**;**i**++){**

**if(**x**==**sov**[**i**]){**

b **=** **true;**

**break;**

**}**

**}**

**if(**b**){**

**return** **true;**

**}**

**else{**

**return** **false;**

**}**

**}**

int elementsBetween**(**int i1**,**int j1**,**int i2**,**int j2**,**int n**){**

int dist**;**

**if(**i1**==**i2**){**

dist **=** abs**(**j2**-**j1**)-**1**;**

**}**

**else{**

**if(**i1**<**i2**){**

dist **=** n**-**j1**-**1**+(**i2**-**i1**-**1**)\***n**+**j2**;**

**}**

**else{**

dist **=** n**-**j2**-**1**+(**i1**-**i2**-**1**)\***n**+**j1**;**

**}**

**}**

**return** dist**;**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,**m**;**

char ch**;**

printf**(**"\nВведите размерность матрицы N = "**);**

scanf**(**"%i"**,&**m**);**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

int a**[**20**][**20**];**

int maxim**=-**999999**,**minim**=**999999**,**sum**,**pr**;**

bool b1**=false,**b2**=false;**

bool fromMaxToMin**=false;**

int kMax**=**0**,**kMin**=**0**,**iMax**,**iMin**,**jMax**,**jMin**;**

**switch(**k**){**

**case** 1**:**

inputMatrKeyB**(**a**,**m**);**

**break;**

**case** 2**:**

inputMatrRand**(**a**,**m**);**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

outputMatr**(**a**,**m**);**

M1**:**

printf**(**"\nВыберите случай:\n1)max(a(i,1),a(i,3),...) в каждой строке если этот эл-т является избыточным(меньше суммы своих правильных делителей)\n2)min(sum(элементов совершенных, Pr(эл-ов, больших суммы своих правильных делителей) по столбцам\n3)Количество элементов расположенных между вторым максимальным от начала матрицы и предпоследним минимальным в каждой зоне\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

printf**(**"\nВывод Ответа:\n"**);**

SetColor**(**1**);**

forI**{**

SetColor**(**1**);**

maxim**=-**999999**;**

forJ**{**

**if(**isExcess**(**a**[**i**][**j**])&&**j**%**2**==**1**){**

SetColor**(**2**);**

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

SetColor**(**1**);**

maxim **=** Max**(**maxim**,**a**[**i**][**j**]);**

**}**

**else{**

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

**}**

**}**

SetColor**(**15**);**

**if(**maxim**!=-**999999**){**

printf**(**" Max = %i"**,**maxim**);**

**}**

**else{**

printf**(**" Таких чисел нет в этой строке!!!"**);**

**}**

printf**(**"\n"**);**

**}**

**break;**

**case** '2'**:**

printf**(**"\nВывод Ответа:\n"**);**

SetColor**(**4**);**

minim**=**999999**;**

forI**{**

forJ**{**

**if(**isSov**(**a**[**i**][**j**])){**

SetColor**(**3**);**

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

SetColor**(**4**);**

**}**

**else** **if(!**isExcess**(**a**[**i**][**j**])){**

SetColor**(**2**);**

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

SetColor**(**4**);**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**15**);**

printf**(**"\n"**);**

forJ**{**

pr **=** 1**;**

sum **=** 0**;**

b1**=**b2**=false;**

forI**{**

**if(**isSov**(**a**[**i**][**j**])){**

sum**+=**a**[**i**][**j**];**

b2 **=** **true;**

**}**

**else** **if(!**isExcess**(**a**[**i**][**j**])&&**a**[**i**][**j**]!=**0**){**

pr**\*=**a**[**i**][**j**];**

b1 **=** **true;**

**}**

**}**

**if(**b1**&&**b2**){**

printf**(**"%4i"**,**Min**(**pr**,**sum**));**

**}**

**else** **if(**b1**){**

printf**(**"%4i"**,**pr**);**

**}**

**else** **if(**b2**){**

printf**(**"%4i"**,**sum**);**

**}**

**else{**

printf**(**" N"**);**

**}**

**}**

**break;**

**case** '3'**:**

printf**(**"\nВывод Ответа:\n"**);**

SetColor**(**1**);**

maxim**=-**999999**;**

k**=**0**;**

minim**=**999999**;**

forI**{**

forJ**{**

**if(**a**[**i**][**j**]>**maxim**){**

maxim **=** a**[**i**][**j**];**

**}**

**if(**a**[**i**][**j**]<**minim**){**

minim **=** a**[**i**][**j**];**

**}**

**}**

**}**

forI**{**

forJ**{**

**if(**a**[**i**][**j**]==**maxim**){**

SetColor**(**2**);**

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

SetColor**(**1**);**

kMax**++;**

**}**

**else** **if(**a**[**i**][**j**]==**minim**){**

SetColor**(**3**);**

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

SetColor**(**1**);**

kMin**++;**

**}**

**else{**

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

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**15**);**

**if(**kMax**<**2**||**kMin**<**2**){**

printf**(**"Матрица не соответствует условию, необходимы два одинаковых минимальных и два максимальных элементов!!\n"**);**

**}**

**else{**

kMax**=**0**;**

forI**{**

forJ**{**

**if(**a**[**i**][**j**]==**maxim**){**

kMax**++;**

**if(**kMax**==**2**){**

jMax **=** j**;**

iMax **=** i**;**

**}**

**}**

**if(**a**[**i**][**j**]==**minim**){**

k**++;**

**if(**k**==**kMin**-**1**){**

iMin **=** i**;**

jMin **=** j**;**

**}**

**}**

**}**

**}**

printf**(**"\nКоличество элементов, расположенных между вторым максимальным и предпоследним минимальным: %i"**,**elementsBetween**(**iMax**,**jMax**,**iMin**,**jMin**,**m**));**

**}**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

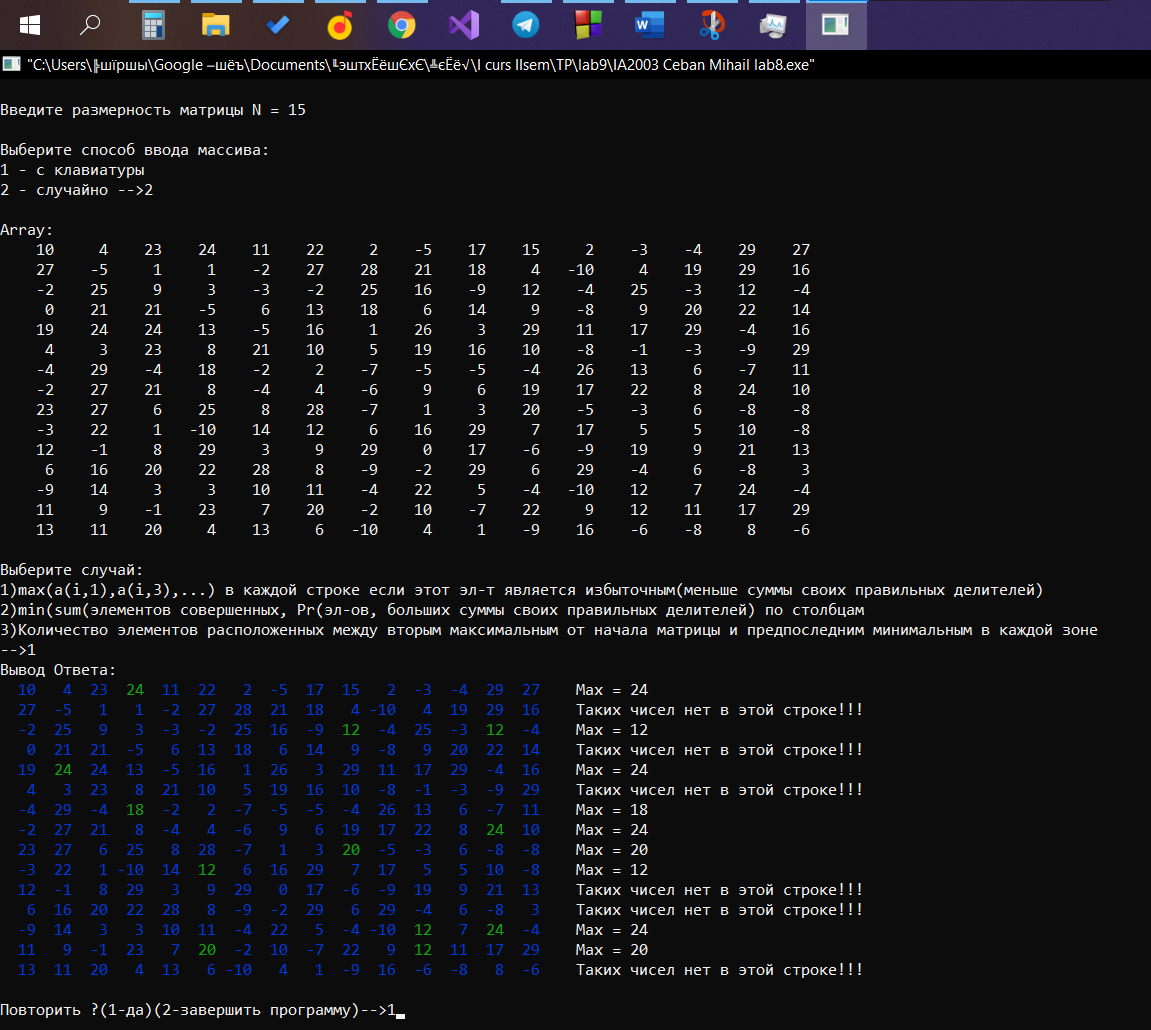
**case** 2**:break;**

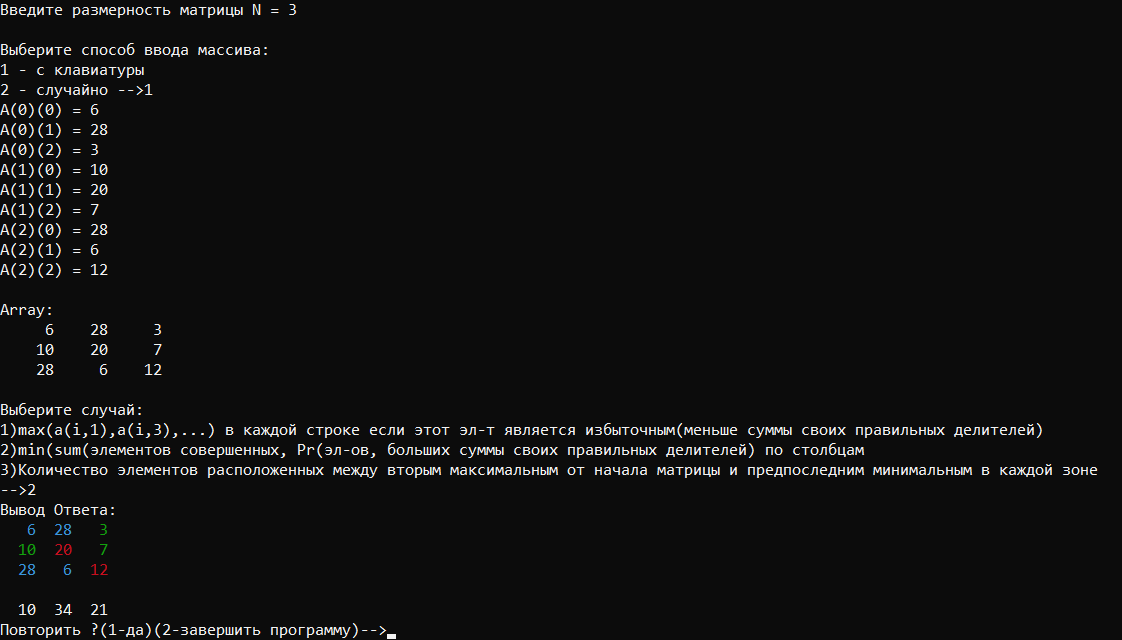
**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

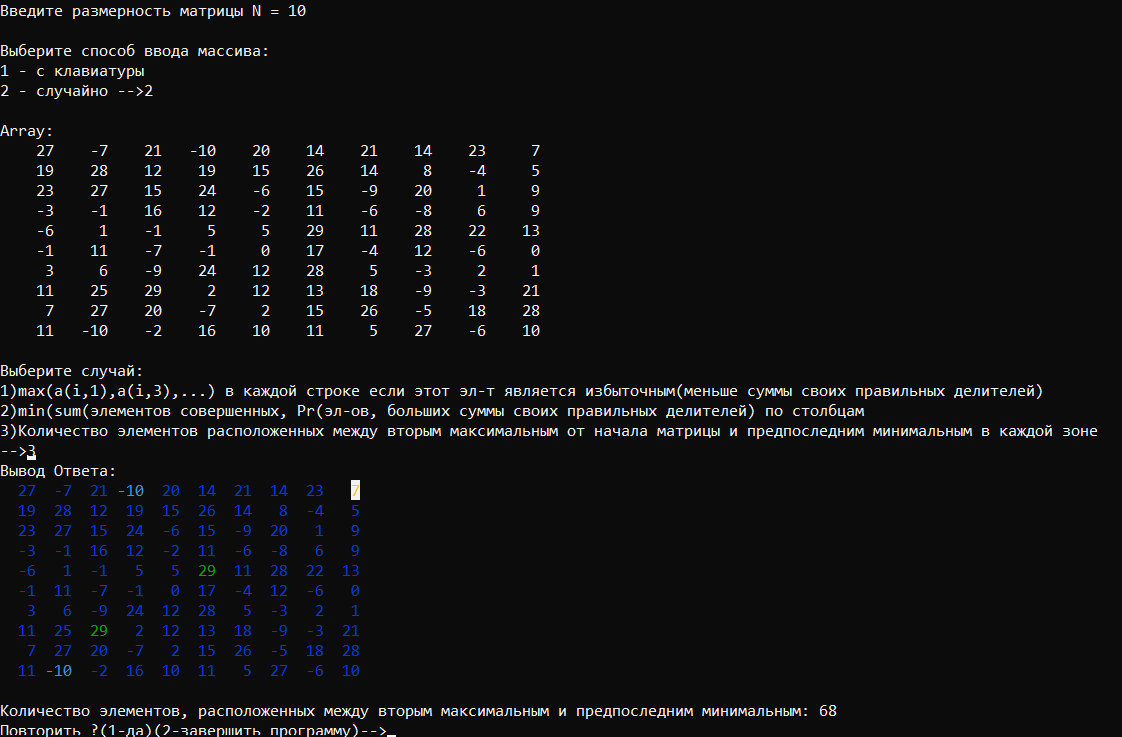
**}**

**}**

**Консоль:**

****

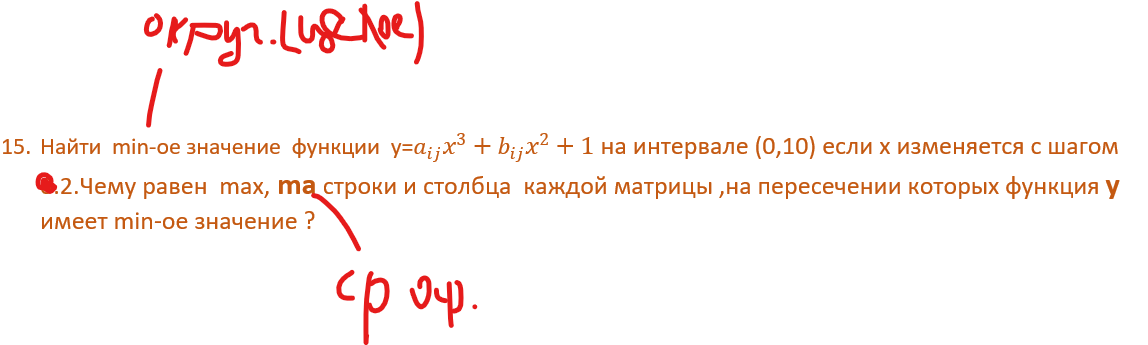
****

****

**Лабораторная работа №10.**

**Задание 1.**

**Условие:**



**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <workWithOMArray.h>

#include <color.h>

#include <iostream>

#define forI for(int i=0;i < n;i++)

#define forJ for(int j=0;j < n;j++)

int y**(**int y**,** int z**,** int x**){**

**return** **(**x**\***x**\***x**\***y**+**x**\***x**\***z**+**1**);**

**}**

int minY**(**int u**,** int v**){**

int i**,**minim**=**999999**;**

**for(**i**=**0**;**i**<=**10**;**i**+=**2**){**

minim **=** Min**(**minim**,**y**(**u**,**v**,**i**));**

**}**

**return** minim**;**

**}**

/\*

Функция, которая получает указатель на двумерный массив и

координаты i,j пересечения строки и столбца, на пересечении

которых функция y имеет минимальное значение.

Фунцкия возвращает минимальное и среднее значение элементов этой строки,

а затем столбца.

\*/

void maxAndSred**(**int **\*\***x**,**int n**,**int yi**,**int yj**,**int **\***maximI**,**int **\***maximJ**,**float **\***sredArI**,**float **\***sredArJ**){**

int summaI**=**0**,**summaJ**=**0**;**

**\***maximI **=** **\***maximJ **=** **-**999999**;**

forI**{**

**\***maximI **=** Max**(\***maximI**,**x**[**yi**][**i**]);**

**\***maximJ **=** Max**(\***maximJ**,**x**[**i**][**yj**]);**

summaI **+=** x**[**yi**][**i**];**

summaJ **+=** x**[**i**][**yj**];**

**}**

**\***sredArI **=** **(**float**)**summaI**/**n**;**

**\***sredArJ **=** **(**float**)**summaJ**/**n**;**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,**n**;**

char ch**;**

printf**(**"\nВведите размерность матрицы N = "**);**

scanf**(**"%i"**,&**n**);**

M2**:**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

int **\*\***a**,** **\*\***b**;**

createDinMatr**(**a**,**n**);**

createDinMatr**(**b**,**n**);**

int ki**[**n**],**kj**[**n**];**

char chA**=**'A'**,**chB**=**'B'**;**

float SredArI**,**SredArJ**;**

int maximI**=-**999999**,**maximJ**=-**999999**;**

**switch(**k**){**

**case** 1**:**

printf**(**"Массив А:\n"**);**

inputMatrKeyB**(**a**,**n**,**chA**);**

printf**(**"Массив B:\n"**);**

inputMatrKeyB**(**b**,**n**,**chB**);**

**break;**

**case** 2**:**

inputMatrRand**(**a**,**n**);**

inputMatrRand**(**b**,**n**);**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M2**;**

**}**

printf**(**"\nArray A:\n"**);**

outputMatr**(**a**,**n**);**

printf**(**"\nArray B:\n"**);**

outputMatr**(**b**,**n**);**

forI**{**

ki**[**i**]** **=** **-**1**;**

kj**[**i**]** **=** **-**1**;**

**}**

printf**(**"\nВывод ответа:\n"**);**

int minYValue**=**999999**;**

SetColor**(**2**);**

forI**{**

forJ**{**

minYValue **=** Min**(**minYValue**,**minY**(**a**[**i**][**j**],**b**[**i**][**j**]));**

**}**

**}**

forI**{**

forJ**{**

**if(**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**){**

SetColor**(**3**);**

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

SetColor**(**2**);**

//запоминаю координаты нужных строк и столбцов

ki**[**i**]** **=** i**;**

kj**[**j**]** **=** j**;**

**}**

**else{**

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

**}**

**}**

printf**(**" | "**);**

forJ**{**

**if(**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**){**

SetColor**(**3**);**

printf**(**"%4i"**,**b**[**i**][**j**]);**

SetColor**(**2**);**

**}**

**else{**

printf**(**"%4i"**,**b**[**i**][**j**]);**

**}**

**}**

printf**(**"\n"**);**

**}**

forI**{**

forJ**{**

**if((**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**)&&(**ki**[**i**]!=-**1**||**kj**[**j**]!=-**1**)){**

SetColor**(**3**);**

printf**(**"\nВывод для массива A:"**);**

SetColor**(**2**);**

maxAndSred**(**a**,**n**,**i**,**j**,&**maximI**,&**maximJ**,&**SredArI**,&**SredArJ**);**

**if(**ki**[**i**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ой строки = %i"**,**i**,**maximI**);**

printf**(**"\nСреднее арифметическое %i-ой строки = %.2f"**,**i**,**SredArI**);**

**}**

**if(**kj**[**j**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ого столбца = %i"**,**j**,**maximJ**);**

printf**(**"\nСреднее арифметическое %i-ого столбца = %.2f"**,**j**,**SredArJ**);**

**}**

SetColor**(**4**);**

printf**(**"\nВывод для массива B:"**);**

SetColor**(**2**);**

maxAndSred**(**b**,**n**,**i**,**j**,&**maximI**,&**maximJ**,&**SredArI**,&**SredArJ**);**

**if(**ki**[**i**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ой строки = %i"**,**i**,**maximI**);**

printf**(**"\nСреднее арифметическое %i-ой строки = %.2f"**,**i**,**SredArI**);**

ki**[**i**]** **=** **-**1**;**

**}**

**if(**kj**[**j**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ого столбца = %i"**,**j**,**maximJ**);**

printf**(**"\nСреднее арифметическое %i-ого столбца = %.2f"**,**j**,**SredArJ**);**

kj**[**j**]** **=** **-**1**;**

**}**

**}**

**}**

**}**

SetColor**(**15**);**

deleteDinMatr**(**a**,**n**);**

deleteDinMatr**(**b**,**n**);**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

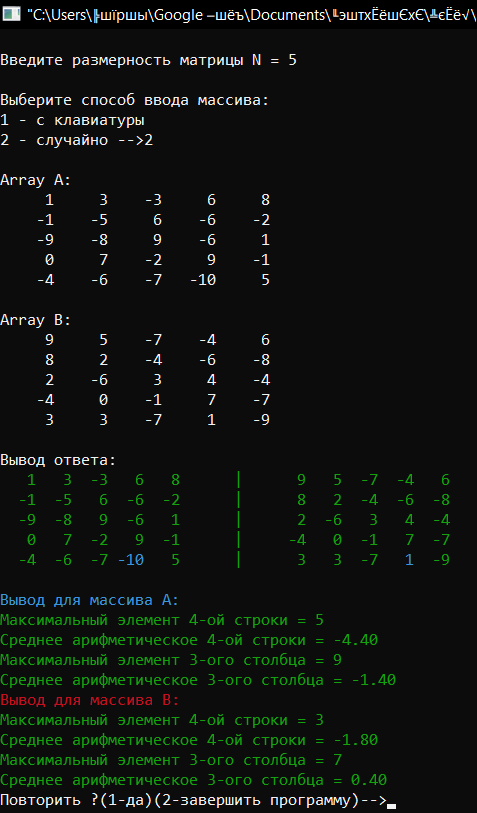
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Лабораторная работа №11.**

**Задание 1.**

**Условие:**

Выполнить 9 лаб.раб. по двумерным массивам с помощью функций и указателей ,передавая весь массив из main() в функцию (MATR\_POINT.cpp) и передав минимум 2 результата из функции в main() аналогично примеру программы point\_6.

**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <workWithOMArray.h>

#include <color.h>

#include <iostream>

#define forI for(int i=0;i < n;i++)

#define forJ for(int j=0;j < n;j++)

int y**(**int y**,** int z**,** int x**){**

**return** **(**x**\***x**\***x**\***y**+**x**\***x**\***z**+**1**);**

**}**

int minY**(**int u**,** int v**){**

int i**,**minim**=**999999**;**

**for(**i**=**0**;**i**<=**10**;**i**+=**2**){**

minim **=** Min**(**minim**,**y**(**u**,**v**,**i**));**

**}**

**return** minim**;**

**}**

/\*

Функция, которая получает указатель на двумерный массив и

координаты i,j пересечения строки и столбца, на пересечении

которых функция y имеет минимальное значение.

Фунцкия возвращает минимальное и среднее значение элементов этой строки,

а затем столбца.

\*/

void maxAndSred**(**int **\*\***x**,**int n**,**int yi**,**int yj**,**int **\***maximI**,**int **\***maximJ**,**float **\***sredArI**,**float **\***sredArJ**){**

int summaI**=**0**,**summaJ**=**0**;**

**\***maximI **=** **\***maximJ **=** **-**999999**;**

forI**{**

**\***maximI **=** Max**(\***maximI**,**x**[**yi**][**i**]);**

**\***maximJ **=** Max**(\***maximJ**,**x**[**i**][**yj**]);**

summaI **+=** x**[**yi**][**i**];**

summaJ **+=** x**[**i**][**yj**];**

**}**

**\***sredArI **=** **(**float**)**summaI**/**n**;**

**\***sredArJ **=** **(**float**)**summaJ**/**n**;**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,**n**;**

char ch**;**

printf**(**"\nВведите размерность матрицы N = "**);**

scanf**(**"%i"**,&**n**);**

M2**:**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

int **\*\***a**,** **\*\***b**;**

createDinMatr**(**a**,**n**);**

createDinMatr**(**b**,**n**);**

int ki**[**n**],**kj**[**n**];**

char chA**=**'A'**,**chB**=**'B'**;**

float SredArI**,**SredArJ**;**

int maximI**=-**999999**,**maximJ**=-**999999**;**

**switch(**k**){**

**case** 1**:**

printf**(**"Массив А:\n"**);**

inputMatrKeyB**(**a**,**n**,**chA**);**

printf**(**"Массив B:\n"**);**

inputMatrKeyB**(**b**,**n**,**chB**);**

**break;**

**case** 2**:**

inputMatrRand**(**a**,**n**);**

inputMatrRand**(**b**,**n**);**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M2**;**

**}**

printf**(**"\nArray A:\n"**);**

outputMatr**(**a**,**n**);**

printf**(**"\nArray B:\n"**);**

outputMatr**(**b**,**n**);**

forI**{**

ki**[**i**]** **=** **-**1**;**

kj**[**i**]** **=** **-**1**;**

**}**

printf**(**"\nВывод ответа:\n"**);**

int minYValue**=**999999**;**

SetColor**(**2**);**

forI**{**

forJ**{**

minYValue **=** Min**(**minYValue**,**minY**(**a**[**i**][**j**],**b**[**i**][**j**]));**

**}**

**}**

forI**{**

forJ**{**

**if(**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**){**

SetColor**(**3**);**

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

SetColor**(**2**);**

//запоминаю координаты нужных строк и столбцов

ki**[**i**]** **=** i**;**

kj**[**j**]** **=** j**;**

**}**

**else{**

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

**}**

**}**

printf**(**" | "**);**

forJ**{**

**if(**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**){**

SetColor**(**3**);**

printf**(**"%4i"**,**b**[**i**][**j**]);**

SetColor**(**2**);**

**}**

**else{**

printf**(**"%4i"**,**b**[**i**][**j**]);**

**}**

**}**

printf**(**"\n"**);**

**}**

forI**{**

forJ**{**

**if((**minY**(**a**[**i**][**j**],**b**[**i**][**j**])==**minYValue**)&&(**ki**[**i**]!=-**1**||**kj**[**j**]!=-**1**)){**

SetColor**(**3**);**

printf**(**"\nВывод для массива A:"**);**

SetColor**(**2**);**

maxAndSred**(**a**,**n**,**i**,**j**,&**maximI**,&**maximJ**,&**SredArI**,&**SredArJ**);**

**if(**ki**[**i**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ой строки = %i"**,**i**,**maximI**);**

printf**(**"\nСреднее арифметическое %i-ой строки = %.2f"**,**i**,**SredArI**);**

**}**

**if(**kj**[**j**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ого столбца = %i"**,**j**,**maximJ**);**

printf**(**"\nСреднее арифметическое %i-ого столбца = %.2f"**,**j**,**SredArJ**);**

**}**

SetColor**(**4**);**

printf**(**"\nВывод для массива B:"**);**

SetColor**(**2**);**

maxAndSred**(**b**,**n**,**i**,**j**,&**maximI**,&**maximJ**,&**SredArI**,&**SredArJ**);**

**if(**ki**[**i**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ой строки = %i"**,**i**,**maximI**);**

printf**(**"\nСреднее арифметическое %i-ой строки = %.2f"**,**i**,**SredArI**);**

ki**[**i**]** **=** **-**1**;**

**}**

**if(**kj**[**j**]!=-**1**){**

printf**(**"\nМаксимальный элемент %i-ого столбца = %i"**,**j**,**maximJ**);**

printf**(**"\nСреднее арифметическое %i-ого столбца = %.2f"**,**j**,**SredArJ**);**

kj**[**j**]** **=** **-**1**;**

**}**

**}**

**}**

**}**

SetColor**(**15**);**

deleteDinMatr**(**a**,**n**);**

deleteDinMatr**(**b**,**n**);**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

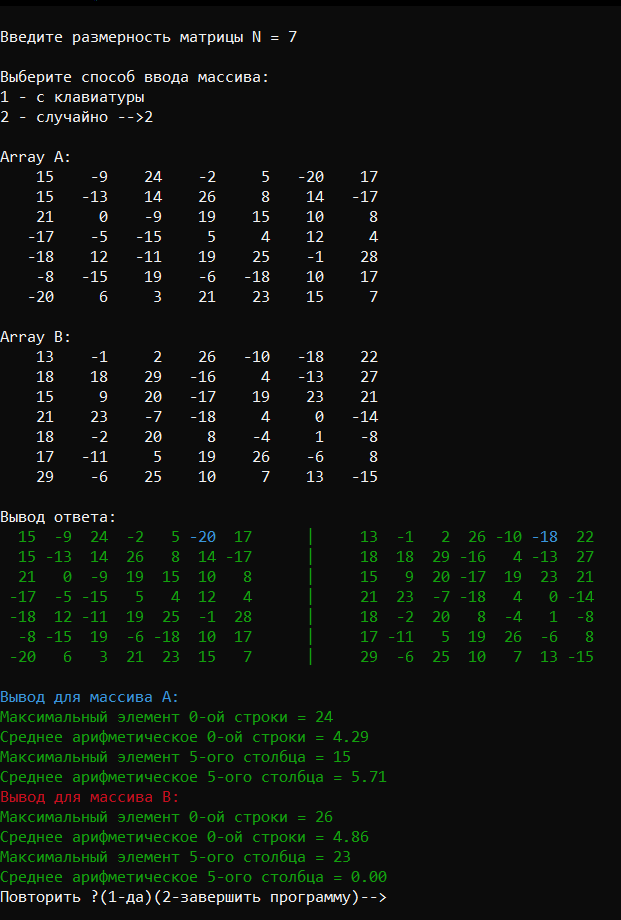
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

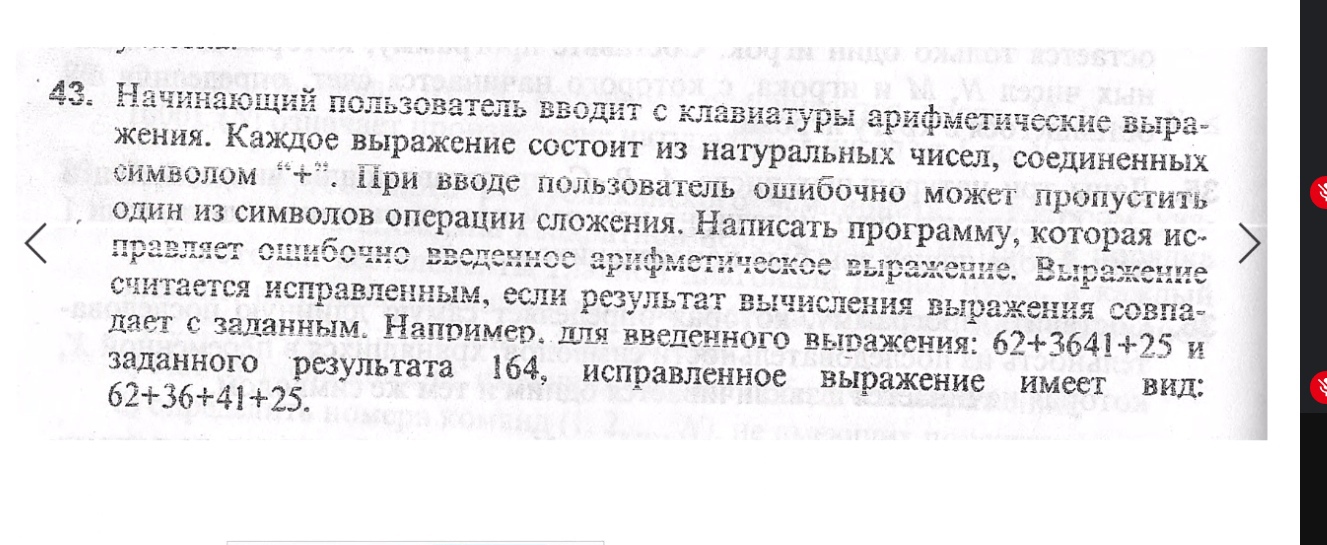
**Лабораторная работа №12.**

**Задание 1.**

**Условие:**

****





**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <color.h>

#include <iostream>

#define forI for(int i=0;str[i]!='\0';i++)

#define forJ for(int j=0;j < n;j++)

int getSize**(**char **\***st**){**

int k**=**0**;**

**for(**int i**=**0**;**st**[**i**]!=**'\0'**;**i**++){**

k**++;**

**}**

**return** k**;**

**}**

int findRez**(**char **\***s**,**int i1**,**int **\***poz1**,**int **\***poz2**){**

int count**=-**1**,**rezult**=**0**,**k**=**0**;**

char num**[**10**];**

**for(**int i**=**0**;**s**[**i**]!=**'\0'**;**i**++){**

**if(**s**[**i**]==**'+'**){**

count**++;**

**}**

**if(!(\***poz1**)&&**i1**!=**0**&&**count**==**i1**-**1**){**

**\***poz1 **=** i**+**1**;**

**}**

**if(!(\***poz2**)&&**count**==**i1**){**

**\***poz2**=**i**;**

**}**

**}**

**if((\***poz2**)==**0**){**

**\***poz2 **=** getSize**(**s**)+**1**;**

**}**

**if(\***poz1**!=**0**){**

**for(**int i**=**0**;**i**<(\***poz1**);**i**++){**

**if(**s**[**i**]==**'+'**){**

num**[**k**]=**'\0'**;**

k**=**0**;**

rezult**+=**atoi**(**num**);**

**}**

**else{**

num**[**k**]** **=** s**[**i**];**

k**++;**

**}**

**}**

**}**

k**=**0**;**

**if((\***poz2**)!=**getSize**(**s**)+**1**){**

**for(**int i **=** **\***poz2**+**1**;**s**[**i**]!=**'\0'**;**i**++){**

**if(**s**[**i**]==**'+'**||**s**[**i**+**1**]==**'\0'**){**

**if(**s**[**i**+**1**]==**'\0'**){**

num**[**k**]=**s**[**i**];**

num**[**k**+**1**]=**'\0'**;**

**}**

**else{**

num**[**k**]=**'\0'**;**

**}**

k**=**0**;**

rezult**+=**atoi**(**num**);**

**}**

**else{**

num**[**k**]** **=** s**[**i**];**

k**++;**

**}**

**}**

**}**

**return** rezult**;**

**}**

void insertSimbol**(**char **\***s**,**int poz**,**char simb**){**

int strSize **=** getSize**(**s**);**

**for(**int i**=**strSize**+**1**;**i**>**poz**;**i**--){**

s**[**i**]** **=** s**[**i**-**1**];**

**}**

s**[**poz**]** **=** simb**;**

**}**

void countSimb**(**char **\***s**,**int **\***gl**,**int **\***sogl**,**int **\***numb**,**int **\***oth**,**int **\***punct**){**

**for(**int i **=** 0**;**s**[**i**]!=**'\0'**;**i**++){**

**if(**s**[**i**]>=**'0'**&&**s**[**i**]<=**'9'**){**

**(\***numb**)++;**

**}**

**else** **if(**s**[**i**]==**'А'**||**s**[**i**]==**'а'**||**s**[**i**]==**'У'**||**s**[**i**]==**'у'**||**s**[**i**]==**'Е'**||**s**[**i**]==**'е'**||**s**[**i**]==**'Э'**||**s**[**i**]==**'э'**||**s**[**i**]==**'О'**||**s**[**i**]==**'о'**||**s**[**i**]==**'Ы'**||**s**[**i**]==**'ы'**||**s**[**i**]==**'Я'**||**s**[**i**]==**'я'**||**s**[**i**]==**'И'**||**s**[**i**]==**'и'**||**s**[**i**]==**'Ю'**||**s**[**i**]==**'ю'**||**s**[**i**]==**'Ё'**||**s**[**i**]==**'ё'**){**

**(\***gl**)++;**

**}**

**else** **if(**s**[**i**]==**'Й'**||**s**[**i**]==**'й'**||**s**[**i**]==**'Ц'**||**s**[**i**]==**'ц'**||**s**[**i**]==**'К'**||**s**[**i**]==**'к'**||**s**[**i**]==**'Н'**||**s**[**i**]==**'н'**||**s**[**i**]==**'Г'**||**s**[**i**]==**'г'**||**s**[**i**]==**'Ш'**||**s**[**i**]==**'ш'**||**s**[**i**]==**'Щ'**||**s**[**i**]==**'щ'**||**s**[**i**]==**'З'**||**s**[**i**]==**'з'**||**s**[**i**]==**'Х'**||**s**[**i**]==**'х'**||**s**[**i**]==**'Ъ'**||**s**[**i**]==**'ъ'**||**s**[**i**]==**'Ф'**||**s**[**i**]==**'ф'**||**s**[**i**]==**'В'**||**s**[**i**]==**'в'**||**s**[**i**]==**'П'**||**s**[**i**]==**'п'**||**s**[**i**]==**'Р'**||**s**[**i**]==**'р'**||**s**[**i**]==**'Л'**||**s**[**i**]==**'л'**||**s**[**i**]==**'Д'**||**s**[**i**]==**'д'**||**s**[**i**]==**'Ж'**||**s**[**i**]==**'ж'**||**s**[**i**]==**'Ч'**||**s**[**i**]==**'ч'**||**s**[**i**]==**'С'**||**s**[**i**]==**'с'**||**s**[**i**]==**'М'**||**s**[**i**]==**'м'**||**s**[**i**]==**'Т'**||**s**[**i**]==**'т'**||**s**[**i**]==**'Ь'**||**s**[**i**]==**'ь'**||**s**[**i**]==**'Б'**||**s**[**i**]==**'б'**){**

**(\***sogl**)++;**

**}**

**else** **if(**ispunct**(**s**[**i**])){**

**(\***punct**)++;**

**}**

**else{**

**(\***oth**)++;**

**}**

**}**

**}**

void findWord**(**char **\***s**,**char **\***w**){**

bool isEqualChar**=false;**

**for(**int i**=**0**;**s**[**i**]!=**'\0'**;**i**++){**

isEqualChar **=** **false;**

**for(**int j**=**0**;**w**[**j**]!=**'\0'**;**j**++){**

**if(**w**[**j**]==**s**[**i**]){**

isEqualChar **=** **true;**

**break;**

**}**

**}**

**if(**isEqualChar**){**

SetColor**(**2**);**

printf**(**"%c"**,**s**[**i**]);**

SetColor**(**15**);**

**}**

**else{**

printf**(**"%c"**,**s**[**i**]);**

**}**

**}**

**}**

void dell**(**char **\***s**,**char **\***s1**,**int i1**,** int j1**,**int length**){**

int dif**=**j1**-**i1**+**1**;**

length**-=**dif**;**

**for(**int i**=**i1**;**s**[**i**+**dif**]!=**'\0'**;**i**++){**

s1**[**i**]** **=** s**[**i**+**dif**];**

**}**

s1**[**length**]** **=** '\0'**;**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

int k**,**n**,**length**=**0**;**

char ch**;**

char str**[**500**],**str1**[**500**],**num1**[**10**],**num2**[**10**];**

int gl**=**0**,**sogl**=**0**,**numb**=**0**,**oth**=**0**,**punct**=**0**;**

int rez**,**pluses**,**poz1**,**poz2**,**rezTest**,**rez2**,**k1**,**k2**,**plusPoint**;**

bool wrongInput**=false,**isAnswer**=false;**

M1**:**

printf**(**"\nВыберите случай:\n1)22.Удалить все символы расположенные между \"\"\n2)15.Найти все буквы введенного слова в введенной строке символов и подсчитать сколько символов каждого типа в этом слове\n3)43.Проверить правильность выражения сложения и по необходимости исправить\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

puts**(**"\nВведите строку на русском -->"**);**

gets**(**str**);**

length**=**0**;**

forI**{**

length**++;**

str1**[**i**]** **=** str**[**i**];**

**}**

str1**[**length**]=**'\0'**;**

forI**{**

**if(**str**[**i**]==**'"'**){**

**for(**int j**=**i**+**1**;**str**[**j**]!=**'\0'**;**j**++){**

**if(**str**[**j**]==**'"'**){**

dell**(**str**,**str1**,**i**,**j**,**length**);**

**}**

**}**

**}**

**}**

printf**(**"\nВывод ответа:\n"**);**

printf**(**"%s"**,**str1**);**

**break;**

**case** '2'**:**

puts**(**"\nВведите строку на русском -->"**);**

gets**(**str**);**

puts**(**"\nВведите слово на русском -->"**);**

gets**(**str1**);**

printf**(**"\nВывод ответа:\n"**);**

findWord**(**str**,**str1**);**

countSimb**(**str1**,&**gl**,&**sogl**,&**numb**,&**oth**,&**punct**);**

printf**(**"\nКоличество цифр = %i"**,**numb**);**

printf**(**"\nКоличесвто гласных = %i"**,**gl**);**

printf**(**"\nКоличество согласных = %i"**,**sogl**);**

printf**(**"\nКоличество символов пунктуации = %i"**,**punct**);**

printf**(**"\nКоличество других символов = %i"**,**oth**);**

**break;**

**case** '3'**:**

puts**(**"\nВведите выражение -->"**);**

gets**(**str**);**

printf**(**"Введите ответ -->"**);**

scanf**(**"%i"**,&**rez**);**

pluses**=**0**;**

forI**{**

**if(**str**[**i**]==**'+'**){**

pluses**++;**

**}**

**else** **if(!**isdigit**(**str**[**i**])){**

wrongInput **=** **true;**

**break;**

**}**

**}**

**if(**wrongInput**){**

printf**(**"\nВы ввели недопустимое выражение!"**);**

**goto** endPr**;**

**}**

k**=**0**;**

rezTest**=**0**;**

forI**{**

**if(**str**[**i**]==**'+'**||**str**[**i**+**1**]==**'\0'**){**

**if(**str**[**i**+**1**]==**'\0'**){**

num1**[**k**]=**str**[**i**];**

num1**[**k**+**1**]=**'\0'**;**

**}**

**else{**

num1**[**k**]=**'\0'**;**

**}**

k**=**0**;**

rezTest**+=**atoi**(**num1**);**

**}**

**else{**

num1**[**k**]** **=** str**[**i**];**

k**++;**

**}**

**}**

**if(**rezTest**==**rez**){**

printf**(**"\nОшибок нет!"**);**

**goto** endPr**;**

**}**

**for(**int i**=**0**;**i**<=**pluses**;**i**++){**

poz1**=**poz2**=**0**;**

rez2 **=** findRez**(**str**,**i**,&**poz1**,&**poz2**);**

**for(**int j**=**0**;**j**<**poz2**-**poz1**-**1**;**j**++){**

k1**=**k2**=**0**;**

**for(**k**=**poz1**;**k**<**poz2**-**j**-**1**;**k**++){**

num1**[**k1**]** **=** str**[**k**];**

k1**++;**

**}**

**for(**k**=**poz2**-**j**-**1**;**k**<**poz2**;**k**++){**

num2**[**k2**]** **=** str**[**k**];**

k2**++;**

**}**

num1**[**k1**]** **=** '\0'**;**

num2**[**k2**]** **=** '\0'**;**

rezTest **=** rez2 **+** atoi**(**num1**)** **+** atoi**(**num2**);**

**if(**rezTest **==** rez**){**

plusPoint **=** poz1**+**k1**;**

isAnswer **=** **true;**

**}**

**if(**isAnswer**){**

**break;**

**}**

**}**

**if(**isAnswer**){**

insertSimbol**(**str**,**plusPoint**,**'+'**);**

printf**(**"\nОтвет: %s"**,**str**);**

**break;**

**}**

**}**

**if(!**isAnswer**){**

printf**(**"\nВы ввели выражение не по условию(пропущен может быть только один '+')!!!"**);**

**}**

**break;**

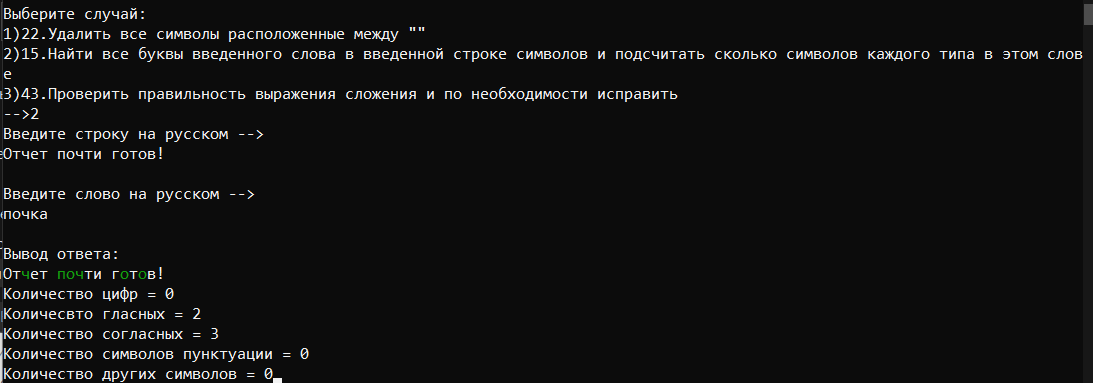
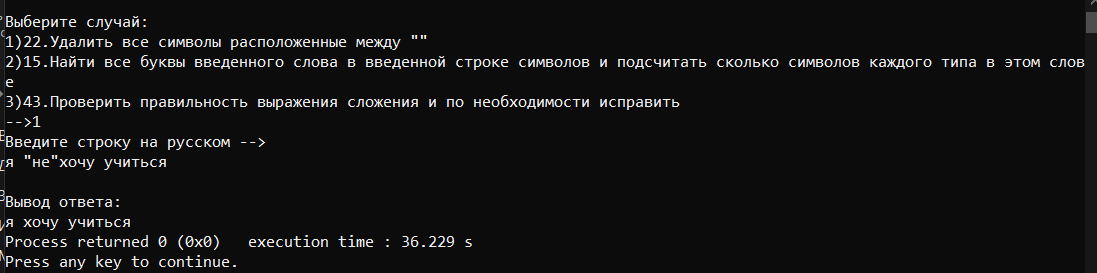
**}**

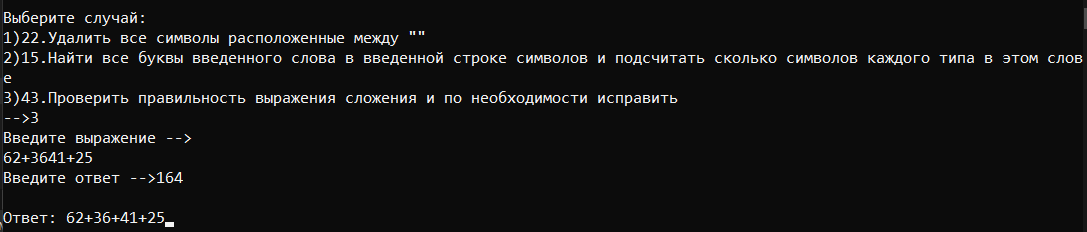
endPr**:**

getche**();**

**}**

**Консоль:**

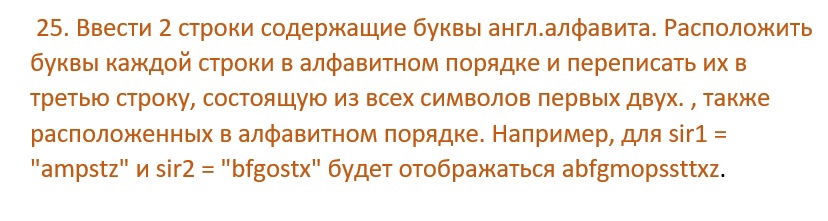
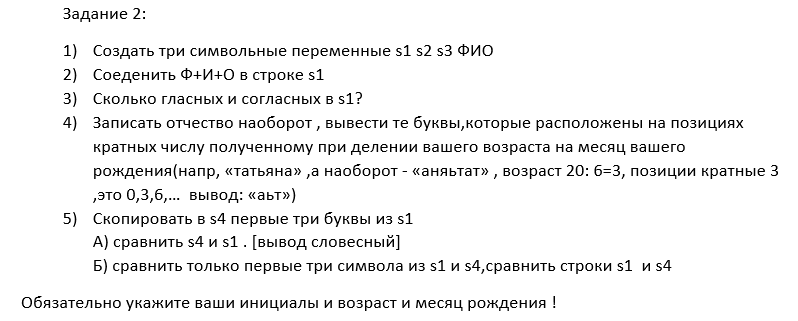
****

****

**Лабораторная работа №13.**

**Задание 1.**

**Условие:**



**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <color.h>

#include <string.h>

#include <iostream>

//В задании с ФИО вводить пробел в конце имени и фамилии!!!

void insertSimbol**(**char **\***s**,**int poz**,**char simb**){**

int strSize **=** strlen**(**s**);**

**for(**int i**=**strSize**+**1**;**i**>**poz**;**i**--){**

s**[**i**]** **=** s**[**i**-**1**];**

**}**

s**[**poz**]** **=** simb**;**

**}**

void delchar**(**char **\***s**,**int poz**){**

int strSize **=** strlen**(**s**);**

**for(**int i**=**poz**;**i**<**strSize**;**i**++){**

s**[**i**]** **=** s**[**i**+**1**];**

**}**

**}**

void delnotalpha**(**char **\***st**){**

forhelp**:**

**for(**int i**=**0**;**i**<**strlen**(**st**);**i**++){**

**if(!**isalpha**(**st**[**i**])){**

delchar**(**st**,**i**);**

**goto** forhelp**;**

**}**

**}**

**}**

void sortalph**(**char **\***st**){**

char ch**;**

**for(**int i**=**strlen**(**st**)-**1**;**i**>=**0**;**i**--){**

**for(**int j**=**0**;**j**<**i**;**j**++){**

**if(**st**[**j**]>**st**[**j**+**1**]){**

ch **=** st**[**j**];**

st**[**j**]** **=** st**[**j**+**1**];**

st**[**j**+**1**]** **=** ch**;**

**}**

**}**

**}**

**}**

void countSimb**(**char **\***s**,**int **\***gl**,**int **\***sogl**,**int **\***numb**,**int **\***oth**,**int **\***punct**){**

char voc**[]** **=** "euioa"**;**

char cons**[]** **=** "qwrtypsdghjklzxcvbnm"**;**

char numbers**[]** **=** "0123456789"**;**

char punctSimb**[]** **=** ".,!?()'':;-"**;**

char other**[]** **=** "@#$%^&\*\_+={[}]></~"**;**

int i**=-**1**;**

char **\***ptr**;**

**while(**voc**[++**i**]!=**'\0'**){**

ptr **=** **NULL;**

ptr **=** strchr**(**s**,**voc**[**i**]);**

**while(**ptr**){**

**(\***gl**)++;**

ptr **=** strchr**(**ptr**+**1**,**voc**[**i**]);**

**}**

**}**

i**=-**1**;**

**while(**cons**[++**i**]!=**'\0'**){**

ptr **=** **NULL;**

ptr **=** strchr**(**s**,**cons**[**i**]);**

**while(**ptr**){**

**(\***sogl**)++;**

ptr **=** strchr**(**ptr**+**1**,**cons**[**i**]);**

**}**

**}**

i**=-**1**;**

**while(**numbers**[++**i**]!=**'\0'**){**

ptr **=** **NULL;**

ptr **=** strchr**(**s**,**numbers**[**i**]);**

**while(**ptr**){**

**(\***numb**)++;**

ptr **=** strchr**(**ptr**+**1**,**numbers**[**i**]);**

**}**

**}**

i**=-**1**;**

**while(**punctSimb**[++**i**]!=**'\0'**){**

ptr **=** **NULL;**

ptr **=** strchr**(**s**,**punctSimb**[**i**]);**

**while(**ptr**){**

**(\***punct**)++;**

ptr **=** strchr**(**ptr**+**1**,**punctSimb**[**i**]);**

**}**

**}**

i**=-**1**;**

**while(**other**[++**i**]!=**'\0'**){**

ptr **=** **NULL;**

ptr **=** strchr**(**s**,**other**[**i**]);**

**while(**ptr**){**

**(\***oth**)++;**

ptr **=** strchr**(**ptr**+**1**,**other**[**i**]);**

**}**

**}**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

int k**,**n**,**length**=**0**;**

char ch**;**

char str**[**100**],**str1**[**100**],**num1**[**10**],**num2**[**10**],**str3**[**100**],**str4**[**100**];**

char **\***ptr1**,\***ptr2**;**

int gl**=**0**,**sogl**=**0**,**numb**=**0**,**oth**=**0**,**punct**=**0**;**

int age**,**month**;**

float nr**;**

M1**:**

printf**(**"\nChoose:\n1)22\n2)15\n3)FIO\n4)25\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

puts**(**"\nInput a string -->"**);**

gets**(**str**);**

strcpy**(**str1**,**str**);**

printf**(**"\nAnswer:\n"**);**

ptr1**=**ptr2**=NULL;**

ptr1 **=** strchr**(**str1**,**'"'**);**

**while(**ptr1**){**

ptr2 **=** strchr**(**ptr1**+**1**,**'"'**);**

**if(**ptr2**){**

str1**[**ptr1**-**str1**]** **=** '\0'**;**

strcat**(**str1**,**ptr2**+**1**);**

ptr1 **=** strchr**(**str1**,**'"'**);**

**}**

**else{**

ptr1 **=** **NULL;**

**}**

**}**

printf**(**"%s"**,**str1**);**

**break;**

**case** '2'**:**

puts**(**"\nInput a string -->"**);**

gets**(**str**);**

puts**(**"\nInput a word -->"**);**

gets**(**str1**);**

printf**(**"\nAnswer:\n"**);**

ptr1 **=** **NULL;**

**for(**int i**=**0**;**str**[**i**]!=**'\0'**;**i**++){**

**for(**int j**=**0**;**str1**[**j**]!=**'\0'**;**j**++){**

**if(**str**[**i**]==**str1**[**j**]){**

SetColor**(**2**);**

**break;**

**}**

**}**

printf**(**"%c"**,**str**[**i**]);**

SetColor**(**15**);**

**}**

countSimb**(**str1**,&**gl**,&**sogl**,&**numb**,&**oth**,&**punct**);**

printf**(**"\nNumbers = %i"**,**numb**);**

printf**(**"\nGlasnie = %i"**,**gl**);**

printf**(**"\nSoglasnie = %i"**,**sogl**);**

printf**(**"\nPunct chars = %i"**,**punct**);**

printf**(**"\nOther simbols = %i"**,**oth**);**

**break;**

**case** '3'**:**

printf**(**"\nInput first name:"**);**

gets**(**str**);**

printf**(**"Input second name:"**);**

gets**(**str1**);**

printf**(**"Input patronymic:"**);**

gets**(**str3**);**

strcat**(**str**,**str1**);**

strcat**(**str**,**str3**);**

printf**(**"%s"**,**str**);**

countSimb**(**str**,&**gl**,&**sogl**,&**numb**,&**oth**,&**punct**);**

printf**(**"\nGlasnie = %i"**,**gl**);**

printf**(**"\nSoglasnie = %i"**,**sogl**);**

printf**(**"\n"**);**

strrev**(**str3**);**

printf**(**"%s"**,**str3**);**

printf**(**"\nYour age:"**);**

scanf**(**"%i"**,&**age**);**

printf**(**"Month of birth:"**);**

scanf**(**"%i"**,&**month**);**

nr **=** age**/**month**;**

printf**(**"Nr - %g"**,**nr**);**

printf**(**"\n%s"**,**str4**);**

**for(**int i**=**0**;**i**<**strlen**(**str3**);**i**++){**

**if(**fmod**(**i**,**nr**)==**0**){**

strncpy**(**str4**,**str3**+**i**,**1**);**

printf**(**"%s"**,**str4**);**

**}**

**}**

printf**(**"\n"**);**

strncpy**(**str4**,**str**,**3**);**

puts**(**str4**);**

printf**(**"\nThe string \"%s\" %c \"%s\""**,**str**,**strcmp**(**str4**,**str**)?**35**:**'='**,**str4**);**

printf**(**"\nThe string \"%s\" %c \"%s\""**,**str**,**strncmp**(**str4**,**str**,**3**)?**35**:**'='**,**str4**);**

**break;**

**case** '4'**:**

puts**(**"\nInput a string -->"**);**

gets**(**str**);**

puts**(**"\nInput another string-->"**);**

gets**(**str1**);**

strlwr**(**str**);**

strlwr**(**str1**);**

delnotalpha**(**str**);**

delnotalpha**(**str1**);**

sortalph**(**str**);**

sortalph**(**str1**);**

printf**(**"\nFirst string sorted:"**);**

printf**(**"\n%s"**,**str**);**

printf**(**"\nSecond string sorted:"**);**

printf**(**"\n%s"**,**str1**);**

strcat**(**str**,**str1**);**

sortalph**(**str**);**

printf**(**"\nThird string sorted:"**);**

printf**(**"\n%s"**,**str**);**

**break;**

**}**

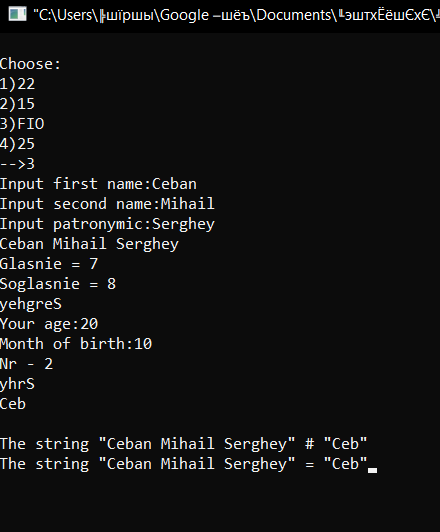
endPr**:**

getche**();**

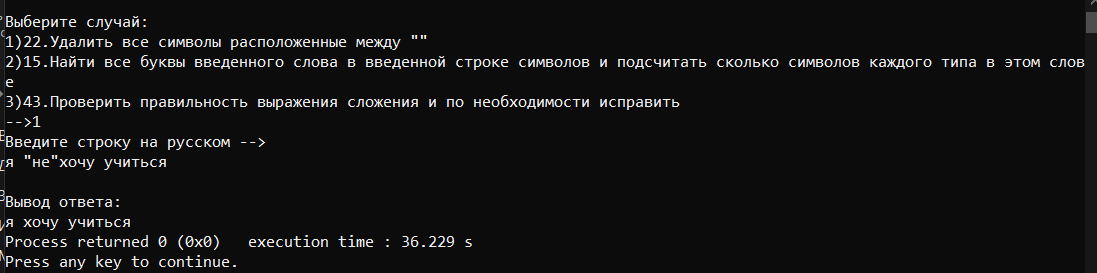
**}**

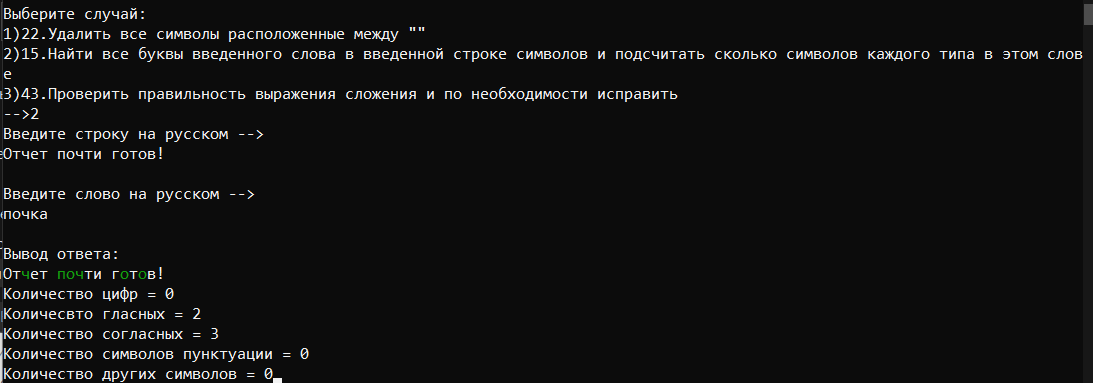
**Консоль:**

**FIO**

****

****

****

****

**Лабораторная работа №14.**

**Задание 1.**

**Условие:**

## 

**Программа:**

#include <stdio.h>

#include <color.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <conio.h>

#include <ctype.h>

struct address**{**

char city**[**20**];**

char street**[**20**];**

int nr**;**

**};**

struct delivery**{**

struct address addr**;**

char tel**[**15**];**

char payment**[**10**];**

**};**

struct charact**{**

int length**;**

int width**;**

int height**;**

**};**

struct tip**{**

char name**[**20**];**

struct charact har**;**

float price**;**

**};**

struct mebel**{**

struct tip type**;**

char manufacturer**[**20**];**

int warranty**;**

struct delivery del**;**

**};**

void input**(**struct mebel x**[],**int n**){**

bool b**=false;**

char ch**[**30**];**

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

printf**(**"\nOrder nr. %i\n"**,**i**+**1**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of furniture:"**);**scanf**(**"%s"**,**x**[**i**].**type**.**name**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**type**.**name**);**j**++){**

b **=** isalpha**(**x**[**i**].**type**.**name**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Length of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**length**);**

printf**(**"Width of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**width**);**

printf**(**"Height of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**height**);**

printf**(**"Price of furniture(in dollars):"**);**scanf**(**"%f"**,&**x**[**i**].**type**.**price**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of manufacturer:"**);**scanf**(**"%s"**,**x**[**i**].**manufacturer**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**manufacturer**);**j**++){**

b **=** isalpha**(**x**[**i**].**manufacturer**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Telephone number:"**);**scanf**(**"%s"**,**x**[**i**].**del**.**tel**);**

printf**(**"Warranty(months):"**);**scanf**(**"%i"**,&**x**[**i**].**warranty**);**

b**=false;**

**while(!**b**){**

printf**(**"Payment method(card or cash):"**);**scanf**(**"%s"**,**x**[**i**].**del**.**payment**);**

**if(!**strcmp**(**x**[**i**].**del**.**payment**,**"card"**)||!**strcmp**(**x**[**i**].**del**.**payment**,**"cash"**)){**

b **=** **true;**

**}**

**else{**

printf**(**"You need to enter words cash or card!\n"**);**

**}**

**}**

printf**(**"Address of delivery:\n"**);**

printf**(**"City:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**city**);**

printf**(**"Street:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**street**);**

printf**(**"Nr. of house:"**);** scanf**(**"%i"**,&**x**[**i**].**del**.**addr**.**nr**);**

**}**

**}**

int main**(){**

SetColor**(**15**);**

int n**=**2**;**

printf**(**"Type number of orders:"**);**scanf**(**"%i"**,&**n**);**

struct mebel a**[**n**];**

input**(**a**,**n**);**

printf**(**"-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

SetColor**(**3**);**

printf**(**"\n| Name | Length | Width | Height | Price | Manufacturer | Warranty | Tel. number | Payment | City | Street |Nr. of house|"**);**

SetColor**(**15**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

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

**{**

printf**(**"\n|%12s|%8i sm.|%7i sm.|%8i sm.|%10.2f$|%16s|%7i months|%15s|%9s|%10s|%14s|%12i|"**,**a**[**i**].**type**.**name**,**a**[**i**].**type**.**har**.**length**,**a**[**i**].**type**.**har**.**width**,**a**[**i**].**type**.**har**.**height**,**

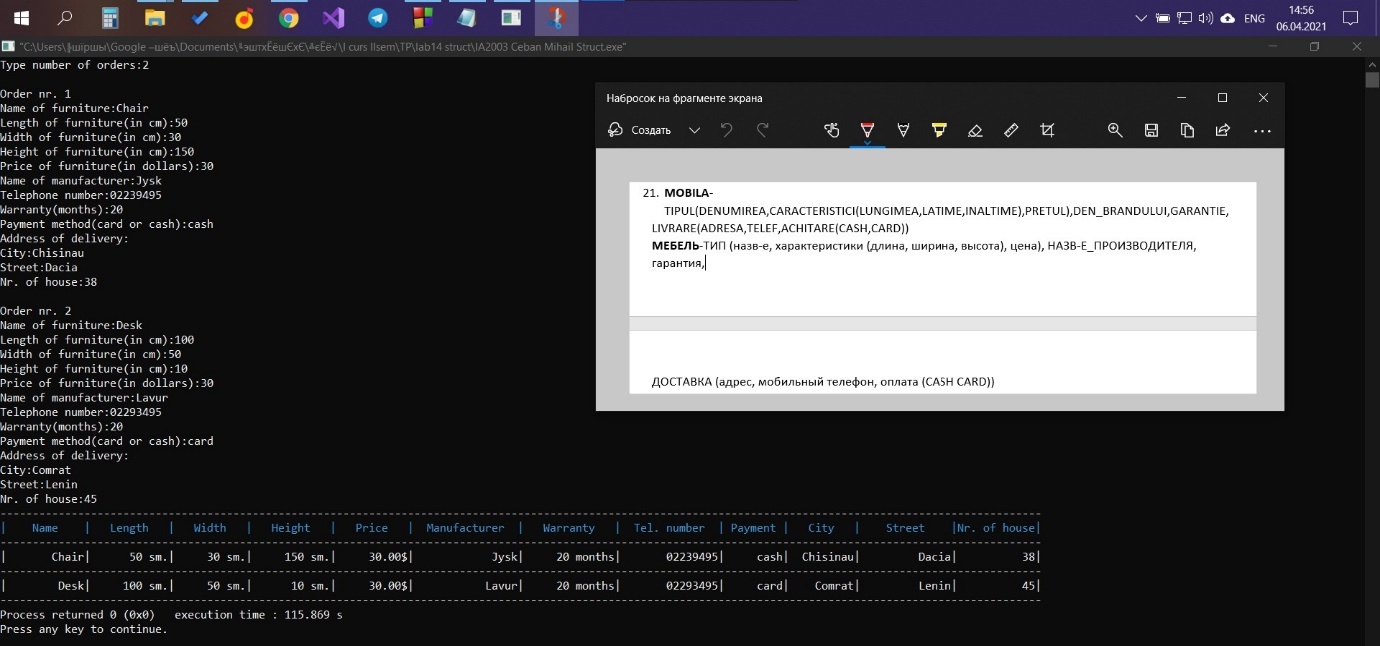
a**[**i**].**type**.**price**,**a**[**i**].**manufacturer**,**a**[**i**].**warranty**,**a**[**i**].**del**.**tel**,**a**[**i**].**del**.**payment**,**a**[**i**].**del**.**addr**.**city**,**a**[**i**].**del**.**addr**.**street**,**a**[**i**].**del**.**addr**.**nr**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**}**

**}**

**Консоль:**

****

**Лабораторная работа №15.**

**Задание 1.**

**Условие:**

## Выполнить меню, в котором работаем со структурой.

## 1)вывод на экран

## 2)поиск

## 3)добавление

## 4)удаление

## 0)выход

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

struct address**{**

char city**[**20**];**

char street**[**20**];**

int nr**;**

**};**

struct delivery**{**

struct address addr**;**

char tel**[**15**];**

char payment**[**10**];**

**};**

struct charact**{**

int length**;**

int width**;**

int height**;**

**};**

struct tip**{**

char name**[**20**];**

struct charact har**;**

float price**;**

**};**

struct mebel**{**

struct tip type**;**

char manufacturer**[**20**];**

int warranty**;**

struct delivery del**;**

**};**

void input**(**struct mebel x**[],**int m**,**int n**){**

bool b**=false;**

**for(**int i**=**m**;**i**<**n**;**i**++){**

printf**(**"\nOrder nr. %i\n"**,**i**+**1**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of furniture:"**);**scanf**(**"%s"**,**x**[**i**].**type**.**name**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**type**.**name**);**j**++){**

b **=** isalpha**(**x**[**i**].**type**.**name**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Length of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**length**);**

printf**(**"Width of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**width**);**

printf**(**"Height of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**height**);**

printf**(**"Price of furniture(in dollars):"**);**scanf**(**"%f"**,&**x**[**i**].**type**.**price**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of manufacturer:"**);**scanf**(**"%s"**,**x**[**i**].**manufacturer**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**manufacturer**);**j**++){**

b **=** isalpha**(**x**[**i**].**manufacturer**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Telephone number:"**);**scanf**(**"%s"**,**x**[**i**].**del**.**tel**);**

printf**(**"Warranty(months):"**);**scanf**(**"%i"**,&**x**[**i**].**warranty**);**

b**=false;**

**while(!**b**){**

printf**(**"Payment method(card or cash):"**);**scanf**(**"%s"**,**x**[**i**].**del**.**payment**);**

**if(!**strcmp**(**x**[**i**].**del**.**payment**,**"card"**)||!**strcmp**(**x**[**i**].**del**.**payment**,**"cash"**)){**

b **=** **true;**

**}**

**else{**

printf**(**"You need to enter words cash or card!\n"**);**

**}**

**}**

printf**(**"Address of delivery:\n"**);**

printf**(**"City:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**city**);**

printf**(**"Street:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**street**);**

printf**(**"Nr. of house:"**);** scanf**(**"%i"**,&**x**[**i**].**del**.**addr**.**nr**);**

**}**

**}**

void output**(**struct mebel x**[],**int m**,** int n**){**

printf**(**"-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

SetColor**(**3**);**

printf**(**"\n| Name | Length | Width | Height | Price | Manufacturer | Warranty | Tel. number | Payment | City | Street |Nr. of house|"**);**

SetColor**(**15**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**for** **(**int i **=** m**;** i **<** n**;** i**++)**

**{**

printf**(**"\n|%12s|%8i sm.|%7i sm.|%8i sm.|%10.2f$|%16s|%7i months|%15s|%9s|%10s|%14s|%12i|"**,**x**[**i**].**type**.**name**,**x**[**i**].**type**.**har**.**length**,**x**[**i**].**type**.**har**.**width**,**x**[**i**].**type**.**har**.**height**,**

x**[**i**].**type**.**price**,**x**[**i**].**manufacturer**,**x**[**i**].**warranty**,**x**[**i**].**del**.**tel**,**x**[**i**].**del**.**payment**,**x**[**i**].**del**.**addr**.**city**,**x**[**i**].**del**.**addr**.**street**,**x**[**i**].**del**.**addr**.**nr**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**}**

**}**

void findEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

bool b**=false;**

printf**(**"\nEnter the name of furniture:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

output**(**x**,**i**,**i**+**1**);**

b **=** **true;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**}**

void delEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

int index**=**0**;**

bool b**=false;**

printf**(**"\nEnter the name of furniture, that you want to delete:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

b **=** **true;**

index **=** i**;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**else{**

**for(**int i**=**index**;**i**<**n**-**1**;**i**++){**

x**[**i**]** **=** x**[**i**+**1**];**

**}**

**}**

**}**

int main**(){**

SetColor**(**15**);**

int n**=**2**,**e**=**3**,**m**=**0**;**

printf**(**"Type number of orders:"**);**scanf**(**"%i"**,&**n**);**

struct mebel a**[**10**];**

struct mebel b**[**10**];**

int k**,**menuCase**;**

M1**:**

printf**(**"1)Example\n2)Input struct.\n-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:**

**for(**int i**=**0**;**i**<**e**;**i**++){**

b**[**i**].**type**.**name **=** **{**'C'**,**'h'**,**'a'**,**'i'**,**'r'**};**

b**[**i**].**type**.**har**.**length **=** 100**;**

b**[**i**].**type**.**har**.**height **=** 150**;**

b**[**i**].**type**.**har**.**width **=** 50**;**

b**[**i**].**type**.**price **=** 15.25**;**

b**[**i**].**manufacturer **=** **{**'J'**,**'y'**,**'s'**,**'k'**};**

b**[**i**].**warranty **=** 13**;**

b**[**i**].**del**.**addr**.**city **=** **{**'C'**,**'h'**,**'i'**,**'s'**,**'i'**,**'n'**,**'a'**,**'u'**};**

b**[**i**].**del**.**addr**.**street **=** **{**'D'**,**'a'**,**'c'**,**'i'**,**'a'**};**

b**[**i**].**del**.**addr**.**nr **=** 24**;**

b**[**i**].**del**.**payment **=** **{**'c'**,**'a'**,**'s'**,**'h'**};**

b**[**i**].**del**.**tel **=** **{**'0'**,**'2'**,**'2'**,**'3'**,**'4'**,**'5'**,**'6'**,**'7'**};**

**}**

b**[**1**].**type**.**name **=** **{**'D'**,**'e'**,**'s'**,**'k'**};**

b**[**2**].**type**.**name **=** **{**'B'**,**'e'**,**'d'**};**

**break;**

**case** 2**:**

input**(**a**,**m**,**n**);**

**break;**

**default:**

printf**(**"Wrong value! Try again!\n"**);**

**goto** M1**;**

**}**

Menu**:**

printf**(**"\nMenu:\n0)Exit\n1)Output struct\n2)Find element by the name\n3)Insert one element to struct\n4)Delete an element in the struct\n-->"**);**

scanf**(**"%i"**,&**menuCase**);**

**switch(**menuCase**){**

**case** 0**:**

**goto** End**;**

**break;**

**case** 1**:**

**if(**k**==**1**){**

output**(**b**,**m**,**e**);**

**}**

**else{**

output**(**a**,**m**,**n**);**

**}**

**break;**

**case** 2**:**

**if(**k**==**1**){**

findEl**(**b**,**e**);**

**}**

**else{**

findEl**(**a**,**n**);**

**}**

**break;**

**case** 3**:**

**if(**k**==**1**){**

input**(**b**,**e**,**e**+**1**);**

e**++;**

**}**

**else{**

input**(**a**,**n**,**n**+**1**);**

n**++;**

**}**

**break;**

**case** 4**:**

**if(**k**==**1**){**

delEl**(**b**,**e**);**

e**--;**

**}**

**else{**

delEl**(**a**,**n**);**

n**--;**

**}**

**break;**

**}**

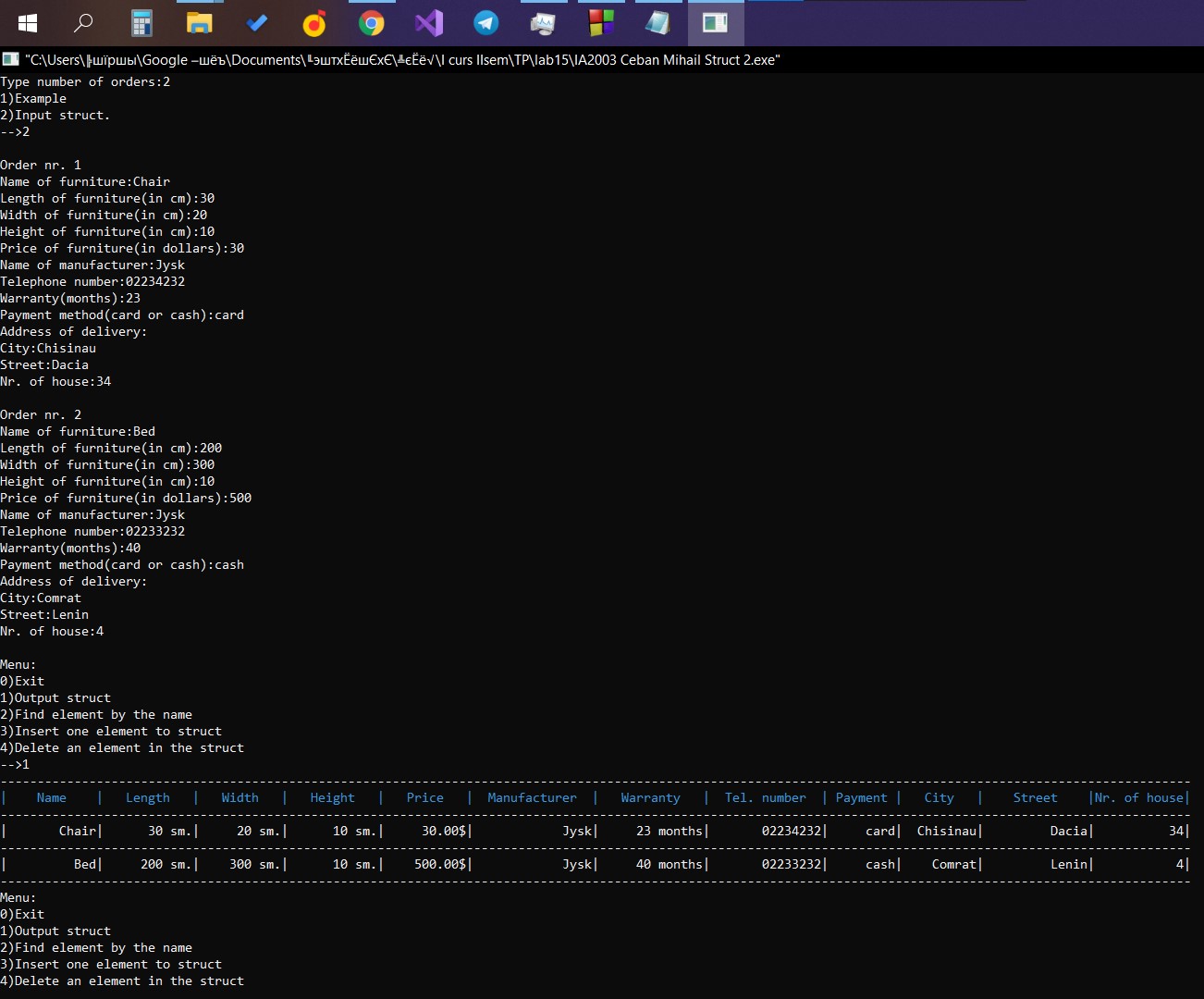
**goto** Menu**;**

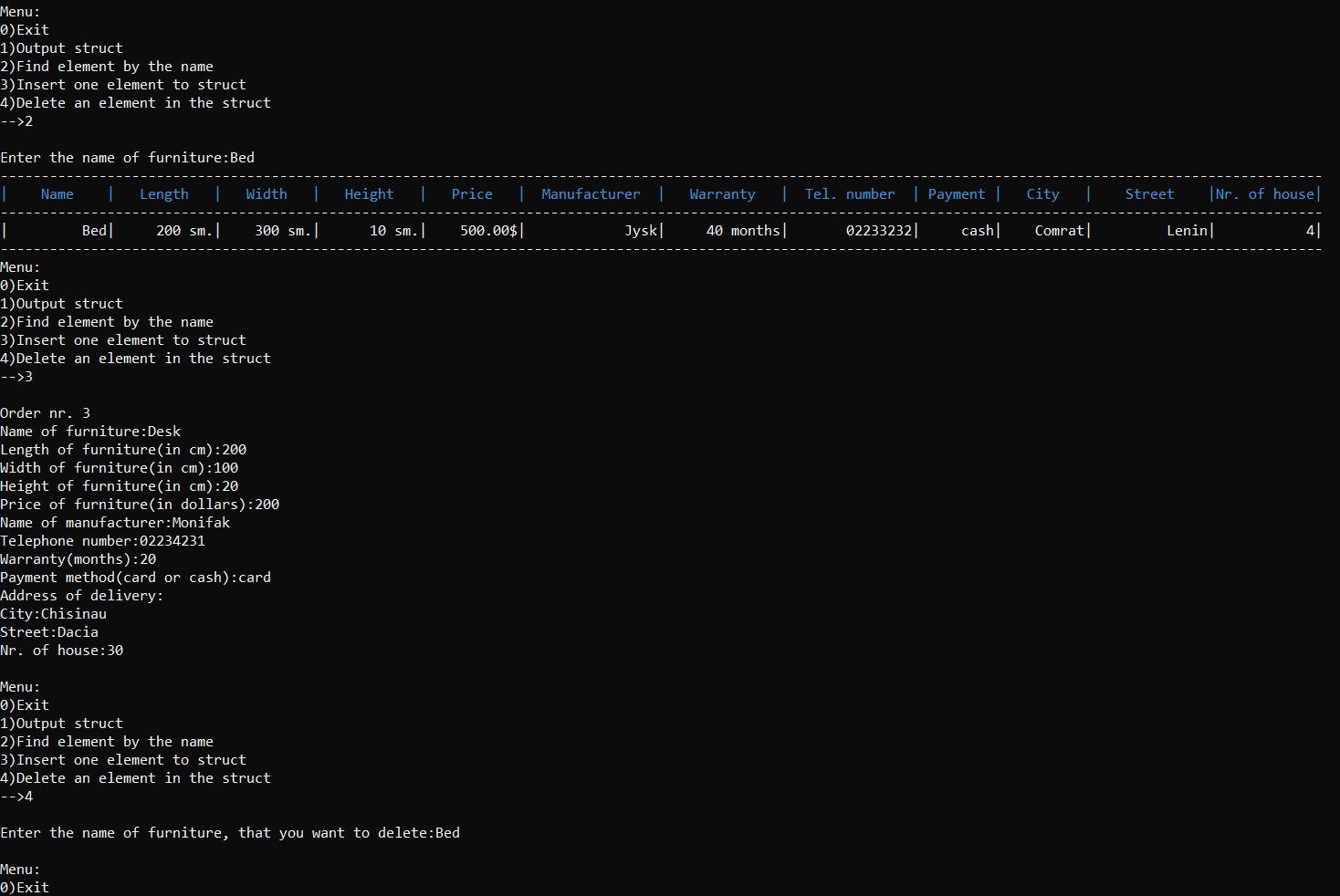
End**:**

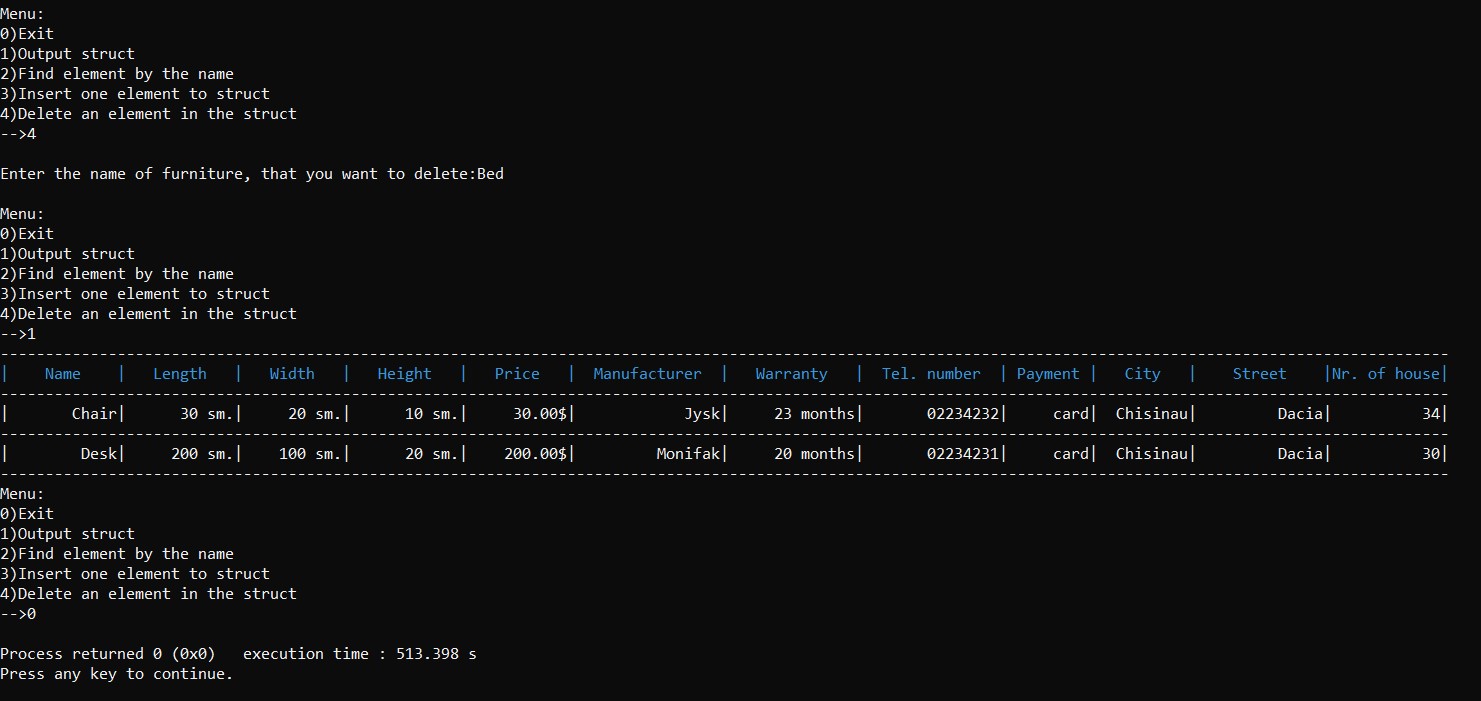
getche**();**

**}**

**Консоль:**

****

****

****

**Лабораторная работа №16.**

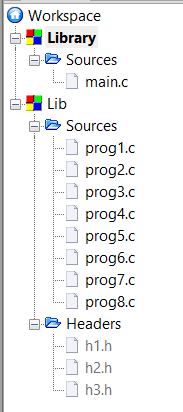
**Задание 1.**

**Условие:**

## Составить проект по условию предыдущих лабораторных работ с одномерным массивом

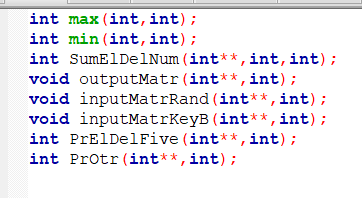
## 

**Программа:**

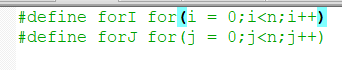


HEADERS

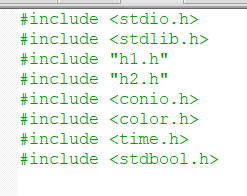
H1.h



H2.h

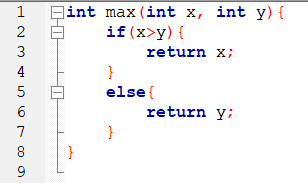


H3.h

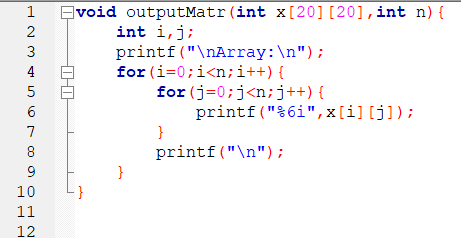


SOURCES

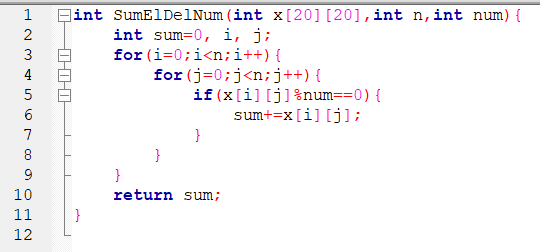
Prog1.c



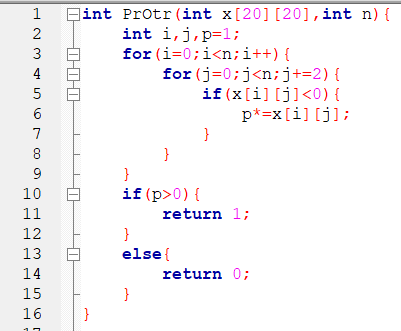
Prog2.c



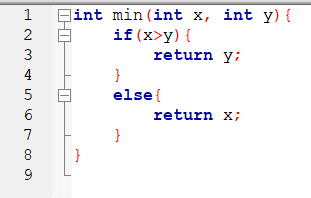
Prog3.c



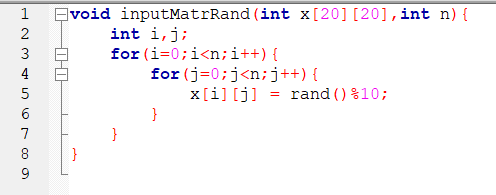
Prog4.c



Prog5.c



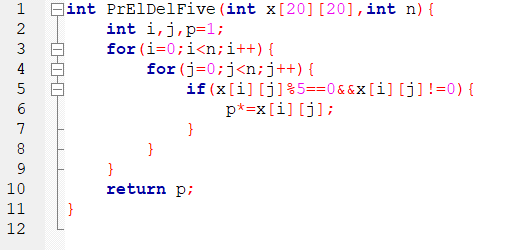
Prog6.c



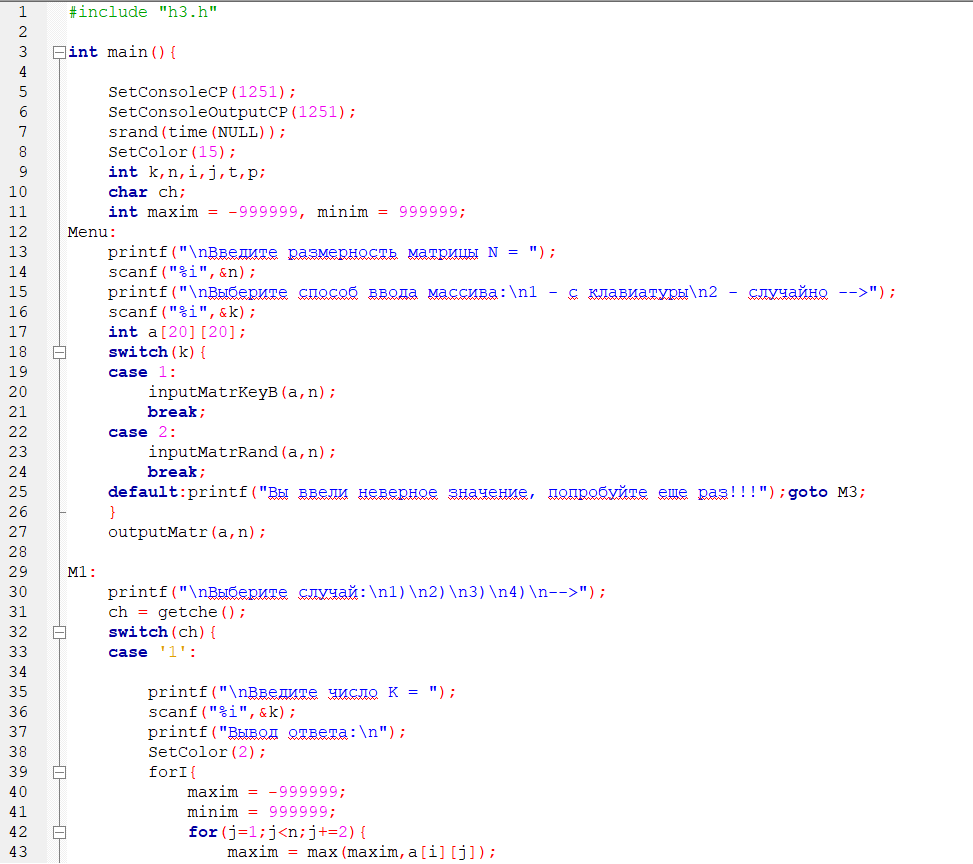
Prog7.c



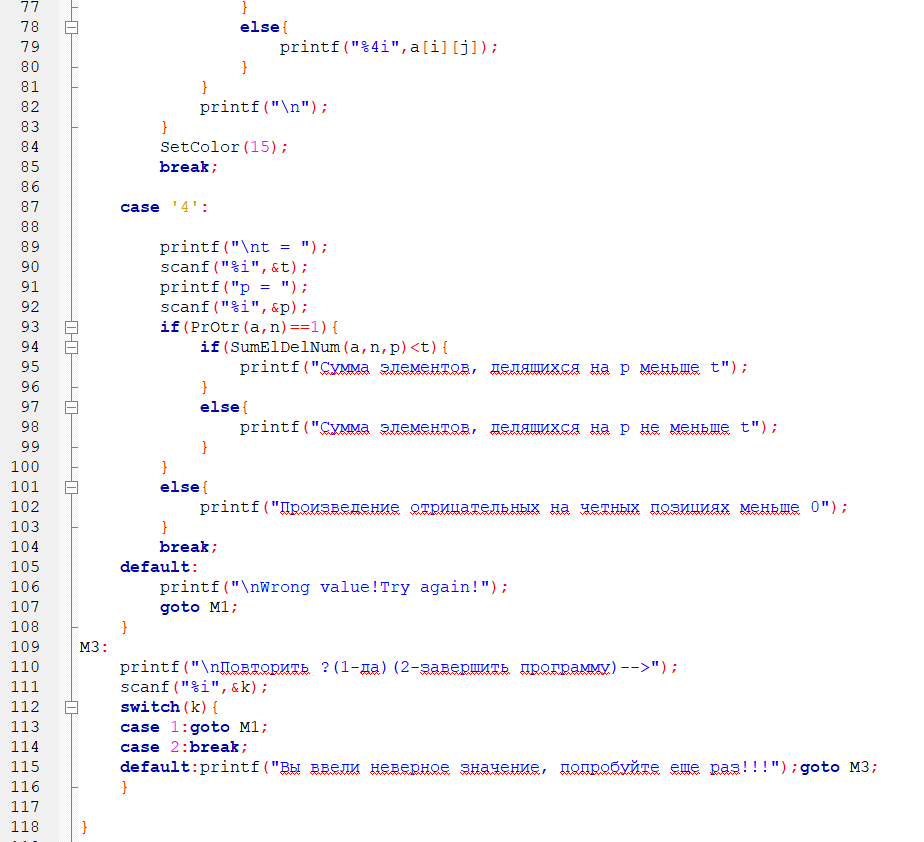
Prog8.c



Main.c

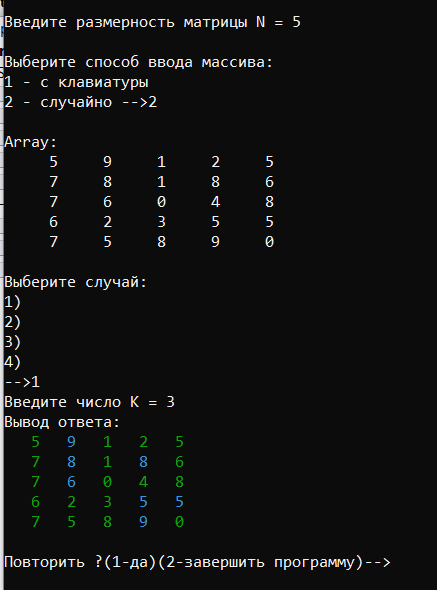




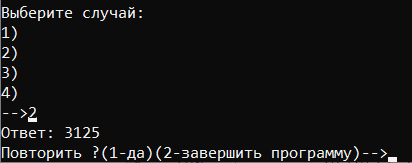


Консоль:

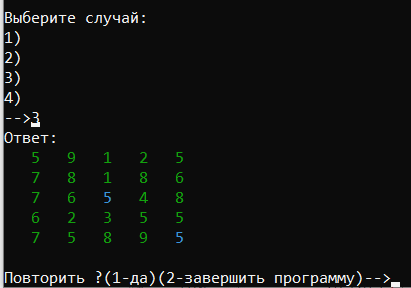
Условие 1:



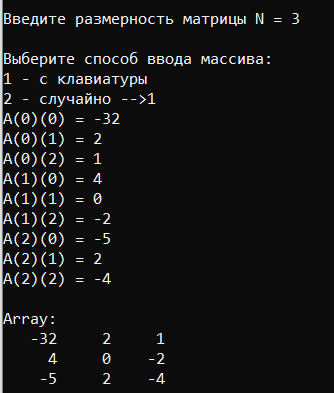
Условие 2:



Условие 3:



Условие 4:



**Задание 2.**

**Условие:**

## 

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

#include <time.h>

void inputKey**(**int x**[],**int n**,** int i**){**

**if** **(**i**<**n**){**

printf**(**"a[%i] = "**,**i**);**

scanf**(**"%i"**,&**x**[**i**]);**

inputKey**(**x**,**n**,**i**+**1**);**

**}**

**}**

void inputRand**(**int x**[],**int n**,** int i**){**

**if** **(**i**<**n**){**

x**[**i**]** **=** rand**()%**200**-**10**;**

inputRand**(**x**,**n**,**i**+**1**);**

**}**

**}**

void out**(**int x**[],**int n**,** int i**){**

**if** **(**i**<**n**){**

printf**(**"%4i"**,**x**[**i**]);**

out**(**x**,**n**,**i**+**1**);**

**}**

**}**

int suma**(**int x**[],**int n**,**int i**,**int y**){**

**if(**i**>=**n**){**

**return** 0**;**

**}**

**else** **if(**x**[**i**]%**y**==**0**){**

**return** x**[**i**]+**suma**(**x**,**n**,**i**+**1**,**y**);**

**}**

**else{**

**return** suma**(**x**,**n**,**i**+**1**,**y**);**

**}**

**}**

int pr**(**int x**[],**int n**,** int i**){**

**if(**i**>=**n**){**

**return** 1**;**

**}**

**else** **if(**x**[**i**]%**5**==**0**&&**x**[**i**]!=**0**){**

**return** x**[**i**]\***pr**(**x**,**n**,**i**+**1**);**

**}**

**else{**

**return** pr**(**x**,**n**,**i**+**1**);**

**}**

**}**

int kol**(**int x**[],**int n**,** int i**){**

**if(**i**<**n**){**

**if(**x**[**i**]==**0**&&**i**%**5**==**0**){**

**return** 1**+**kol**(**x**,**n**,**i**+**1**);**

**}**

**else{**

**return** kol**(**x**,**n**,**i**+**1**);**

**}**

**}**

**else{**

**return** 0**;**

**}**

**}**

int prOtr**(**int x**[],**int n**,**int i**){**

**if(**i**<**n**){**

**if(**x**[**i**]<**0**&&**i**%**2**==**0**){**

**return** x**[**i**]\***prOtr**(**x**,**n**,**i**+**1**);**

**}**

**else{**

**return** prOtr**(**x**,**n**,**i**+**1**);**

**}**

**}**

**return** 1**;**

**}**

int Max**(**int o**,**int y**){**

**if(**o**>**y**){**

**return** o**;**

**}**

**else** **return** y**;**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,** n**=**4**,**a**[**50**],**i**=**0**,**m**,**p**;**

printf**(**"Type dimension of array N = "**);**

scanf**(**"%i"**,&**n**);**

printf**(**"Input with keyboard or random?(1 | 2)\n-->"**);**

scanf**(**"%i"**,&**k**);**

**if(**k**==**1**){**

inputKey**(**a**,**n**,**0**);**

**}**

**else{**

inputRand**(**a**,**n**,**0**);**

**}**

out**(**a**,**n**,**0**);**

M1**:**

printf**(**"\nChoose the case:\n2)\n3)\n4)\n-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 2**:**

printf**(**"Answer: %i"**,**Max**(**suma**(**a**,**n**,**0**,**3**),**pr**(**a**,**n**,**0**)));**

**break;**

**case** 3**:**

printf**(**"Answer: %i"**,**kol**(**a**,**n**,**0**));**

**break;**

**case** 4**:**

printf**(**"m = "**);**

scanf**(**"%i"**,&**m**);**

printf**(**"p = "**);**

scanf**(**"%i"**,&**p**);**

**if(**prOtr**(**a**,**n**,**0**)>**0**){**

**if(**suma**(**a**,**n**,**i**,**m**)<**p**){**

printf**(**"Сумма элементов, делящихся на m меньше p"**);**

**}**

**else{**

printf**(**"Сумма элементов, делящихся на m не меньше p"**);**

**}**

**}**

**else{**

printf**(**"Произведение отрицательных на четных позициях меньше 0"**);**

**}**

**break;**

**default:**

printf**(**"Wrong value!Try again!"**);**

**goto** M1**;**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

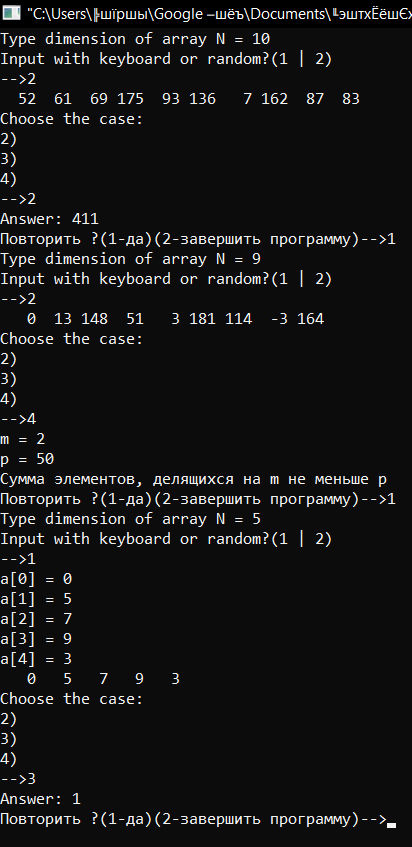
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Задание 3.**

**Условие:**

## 

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

#include <time.h>

#include <math.h>

void out**(**float x**[],**int n**,** int i**){**

**if** **(**i**<**n**){**

printf**(**"%10g"**,**x**[**i**]);**

out**(**x**,**n**,**i**+**1**);**

**}**

**}**

void method1**(**float x**[],**int n**,** int i**){**

**if(**i**<**n**&&**i**>**1**){**

x**[**i**]** **=** x**[**i**-**1**]/**2.0**+**3**\***x**[**i**-**2**]/**4.0**;**

method1**(**x**,**n**,**i**+**1**);**

**}**

**else** **if(**i**<**2**){**

x**[**0**]=**0**;**

x**[**1**]=**5**/**8.0**;**

method1**(**x**,**n**,**2**);**

**}**

**}**

void method2**(**float x**[],**int n**,** int i**){**

**if(**i**<**n**&&**i**>**1**){**

x**[**i**]** **=** **(**i**+**1**)\***x**[**i**-**2**];**

method2**(**x**,**n**,**i**+**1**);**

**}**

**else** **if(**i**<**2**){**

x**[**0**]=**1.0**;**

x**[**1**]=**0.3**;**

method2**(**x**,**n**,**2**);**

**}**

**}**

void method3**(**float x**[],**int n**,** int i**){**

**if(**i**<**n**&&**i**>**2**){**

x**[**i**]** **=** **(**i**+**3**)\*(**x**[**i**-**1**]-**1**)+(**i**+**4**)\***x**[**i**-**3**];**

method3**(**x**,**n**,**i**+**1**);**

**}**

**else** **if(**i**<**2**){**

x**[**0**]=**x**[**1**]=**x**[**2**]=**1**;**

method3**(**x**,**n**,**3**);**

**}**

**}**

float suma1**(**float x**[],**int n**,**int i**,**float o**){**

**if(**i**<**n**){**

**return** x**[**i**]/**o**+**suma1**(**x**,**n**,**i**+**1**,**o**);**

**}**

**else{**

**return** 0**;**

**}**

**}**

float pr1**(**float x**[],**int n**,**int i**){**

**if(**i**<**n**){**

**return** log**(**1**+**x**[**i**])\***pr1**(**x**,**n**,**i**+**1**);**

**}**

**else{**

**return** 1**;**

**}**

**}**

float suma2**(**float x**[],**int n**,**int i**){**

**if(**i**<**n**-**1**){**

**return** x**[**i**]\***x**[**i**+**1**]+**suma2**(**x**,**n**,**i**+**1**);**

**}**

**else{**

**return** 0**;**

**}**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,** n**=**4**,**i**=**0**,**m**,**p**;**

float a**[**50**],**o**;**

printf**(**"Type dimension of array N = "**);**

scanf**(**"%i"**,&**n**);**

M1**:**

printf**(**"\nChoose the case:\n1)\n2)\n3)\n-->"**);**

scanf**(**"%i"**,&**p**);**

**switch(**p**){**

**case** 1**:**

method1**(**a**,**n**,**0**);**

out**(**a**,**n**,**i**);**

M2**:**

printf**(**"\n(o!=0)o = "**);**

scanf**(**"%f"**,&**o**);**

printf**(**"\n(m<n)m = "**);**

scanf**(**"%i"**,&**m**);**

**if(**m**<**n**&&**m**>**0**&&**o**!=**0**){**

printf**(**"Sum from 0 to m of x[i]/o = %.2f"**,**suma1**(**a**,**m**,**0**,**o**));**

**}**

**else{**

printf**(**"Error!Try again!"**);**

**goto** M2**;**

**}**

**break;**

**case** 2**:**

method2**(**a**,**n**,**0**);**

out**(**a**,**n**,**i**);**

M4**:**

printf**(**"\n(k<n)k = "**);**

scanf**(**"%i"**,&**k**);**

**if(**k**<**n**&&**k**>**0**){**

printf**(**"P of log(1+x[i]) = %.2f"**,**pr1**(**a**,**k**,**0**));**

**}**

**else{**

printf**(**"Error!Try again!"**);**

**goto** M4**;**

**}**

**break;**

**case** 3**:**

method3**(**a**,**n**,**0**);**

out**(**a**,**n**,**i**);**

printf**(**"\nSum(x[1]\*x[2]+x[2]\*x[3]+...) = %.2f"**,**suma2**(**a**,**n**,**0**));**

**break;**

**default:**

printf**(**"Wrong value!Try again!"**);**

**goto** M1**;**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

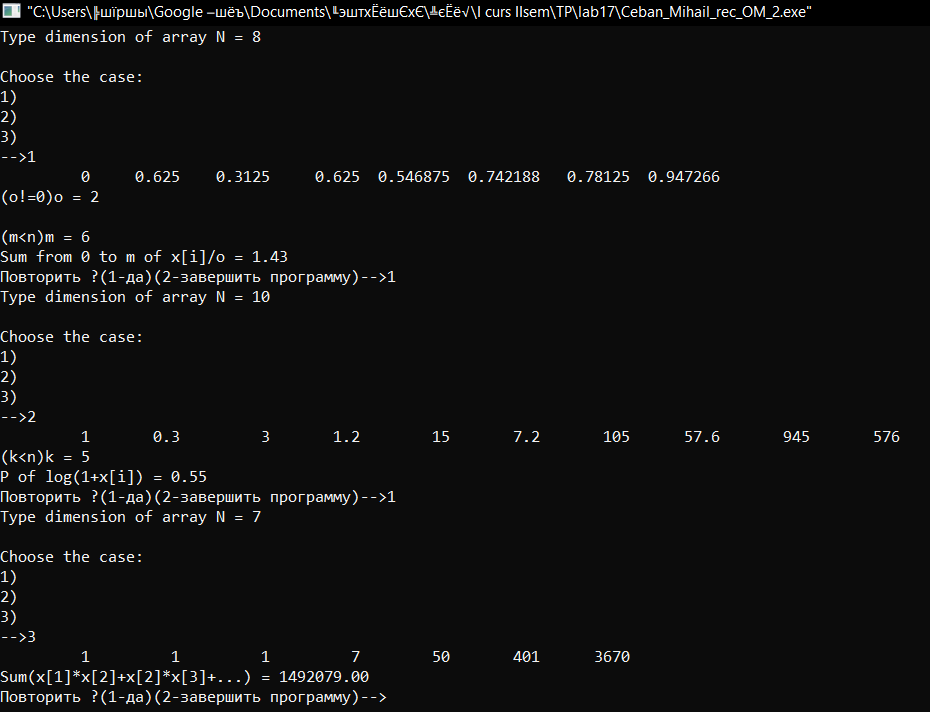
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Задание 4.**

**Условие:**

## 

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

#include <time.h>

void inputKey**(**int **\*\***x**,**int n**,**int i**,**int j**){**

**if** **(**i**<**n**){**

**if(**j**<**n**){**

printf**(**"a[%i,%i] = "**,**i**,**j**);**

scanf**(**"%i"**,&**x**[**i**][**j**]);**

inputKey**(**x**,**n**,**i**,**j**+**1**);**

**}**

**else{**

inputKey**(**x**,**n**,**i**+**1**,**0**);**

**}**

**}**

**}**

void inputRand**(**int **\*\***x**,**int n**,** int i**,**int j**){**

**if** **(**i**<**n**){**

**if(**j**<**n**){**

x**[**i**][**j**]** **=** rand**()%**50**-**10**;**

inputRand**(**x**,**n**,**i**,**j**+**1**);**

**}**

**else{**

inputRand**(**x**,**n**,**i**+**1**,**0**);**

**}**

**}**

**}**

void out**(**int **\*\***x**,**int n**,** int i**,**int j**){**

**if** **(**i**<**n**){**

**if(**j**<**n**){**

printf**(**"%5i"**,**x**[**i**][**j**]);**

out**(**x**,**n**,**i**,**j**+**1**);**

**}**

**else{**

printf**(**"\n"**);**

out**(**x**,**n**,**i**+**1**,**0**);**

**}**

**}**

**}**

int Min**(**int o**,** int y**){**

**if(**o**>**y**){**

**return** y**;**

**}**

**else{**

**return** o**;**

**}**

**}**

int Max**(**int o**,** int y**){**

**if(**o**>**y**){**

**return** o**;**

**}**

**else{**

**return** y**;**

**}**

**}**/\*

int minim(int \*x,int n, int i, int minPoint){

if(i<n-1){

if(x[i]>0){

minPoint = Min(x[i],minim(x,n,i+1,minPoint));

}

else{

minim(x,n,i+1,x[i+1]);

}

}

else{

return minPoint;

}

}\*/

int minim**(**int **\***x**,**int n**,** int i**,**int minI**,** int minPoint**){**

**if(**i**==**n**){**

**return** minI**;**

**}**

**if(**minPoint**>**x**[**i**]&&**x**[**i**]>=**0**){**

minPoint **=** x**[**i**];**

minI **=** i**;**

**}**

i**++;**

**return** minim**(**x**,**n**,**i**,**minI**,**minPoint**);**

**}**

int maxim**(**int **\***x**,**int n**,** int i**,**int maxI**,** int maxPoint**){**

**if(**i**==**n**){**

**return** maxI**;**

**}**

**if(**maxPoint**<**x**[**i**]){**

maxPoint **=** x**[**i**];**

maxI **=** i**;**

**}**

i**++;**

**return** maxim**(**x**,**n**,**i**,**maxI**,**maxPoint**);**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,** n**=**4**,\*\***a**,\***minPoints**,**i**=**0**,**m**,**p**;**

bool b**;**

printf**(**"Type dimension of array N = "**);**

scanf**(**"%i"**,&**n**);**

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

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

a**[**i**]** **=** **new** int **[**n**];**

**}**

minPoints **=** **new** int **[**n**];**

printf**(**"Input with keyboard or random?(1 | 2)\n-->"**);**

scanf**(**"%i"**,&**k**);**

**if(**k**==**1**){**

inputKey**(**a**,**n**,**0**,**0**);**

**}**

**else{**

inputRand**(**a**,**n**,**0**,**0**);**

**}**

printf**(**"Array A:\n"**);**

out**(**a**,**n**,**0**,**0**);**

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

minPoints**[**i**]** **=** minim**(**a**[**i**],**n**,**0**,**0**,**999999**);**

**}**

printf**(**"\n"**);**

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

printf**(**"%4i"**,**minPoints**[**i**]);**

**}**

printf**(**"\n"**);**

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

printf**(**"%6i"**,**maxim**(**a**[**i**],**n**,**1**,**0**,**a**[**i**][**0**]));**

**}**

printf**(**"\nAnswer:\n"**);**

SetColor**(**2**);**

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

b **=** **false;**

**if(**maxim**(**a**[**i**],**n**,**1**,**0**,**a**[**i**][**0**])==**i**){**

p **=** a**[**i**][**maxim**(**a**[**i**],**n**,**1**,**0**,**a**[**i**][**0**])];**

a**[**i**][**maxim**(**a**[**i**],**n**,**1**,**0**,**a**[**i**][**0**])]** **=** a**[**i**][**minPoints**[**i**]];**

a**[**i**][**minPoints**[**i**]]** **=** p**;**

b **=** **true;**

**}**

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

**if((**j**==**minPoints**[**i**]||**j**==**i**)&&(**b**)){**

SetColor**(**3**);**

printf**(**"%5i"**,**a**[**i**][**j**]);**

SetColor**(**2**);**

**}**

**else{**

printf**(**"%5i"**,**a**[**i**][**j**]);**

**}**

**}**

printf**(**"\n"**);**

**}**

SetColor**(**15**);**

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

**delete[]** a**[**i**];**

**}**

**delete[]** a**;**

**delete[]** minPoints**;**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

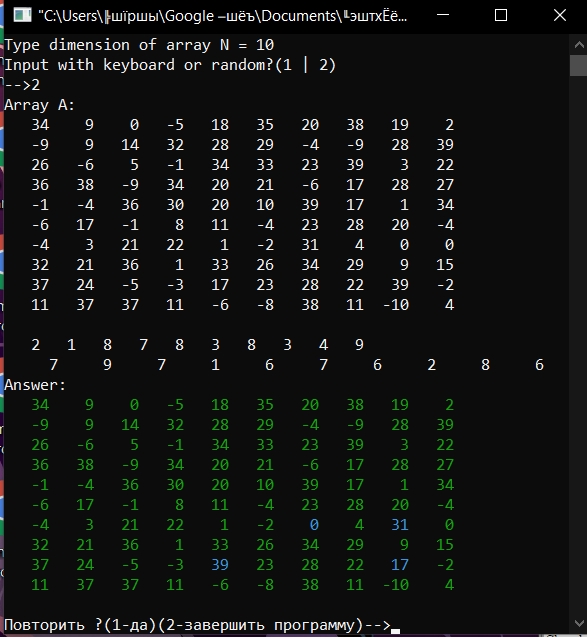
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Задание 5.**

**Условие:**

## 

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

#include <time.h>

#include <math.h>

void inputKey**(**int **\***x**,**int n**,**int j**,**char ch**){**

**if(**j**<**n**){**

printf**(**"%c[%i] = "**,**ch**,**j**);**

scanf**(**"%i"**,&**x**[**j**]);**

inputKey**(**x**,**n**,**j**+**1**,**ch**);**

**}**

**}**

void inputRand**(**int **\***x**,**int n**,** int j**){**

**if(**j**<**n**){**

x**[**j**]** **=** rand**()%**50**-**10**;**

inputRand**(**x**,**n**,**j**+**1**);**

**}**

**}**

void out**(**int **\***x**,**int n**,** int j**){**

**if(**j**<**n**){**

printf**(**"%5i"**,**x**[**j**]);**

out**(**x**,**n**,**j**+**1**);**

**}**

**}**

int sum**(**int **\***x**,**int **\***y**,**int n**,**int j**){**

**if(**j**==**n**-**1**){**

**return** x**[**j**]\***y**[**j**]+**j**;**

**}**

**return** **(**x**[**j**]\***y**[**j**]+**j**)\***sum**(**x**,**y**,**n**,**j**+**1**);**

**}**

float sum1**(**int **\***x**,**int **\***y**,**int n**,**int i**){**

**if(**i**==**n**){**

**return** n**/**sum**(**x**,**y**,**i**,**0**);**

**}**

**return** **(**float**)(**i**+**1**)/**sum**(**x**,**y**,**i**+**1**,**0**)+**sum1**(**x**,**y**,**n**,**i**+**1**);**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

SetColor**(**15**);**

Menu**:**

int k**,** n**,\***a**,\***b**,**i**=**0**,**m**,**p**;**

printf**(**"Type dimension of array N = "**);**

scanf**(**"%i"**,&**n**);**

a **=** **new** int **[**n**];**

b **=** **new** int **[**n**];**

printf**(**"Input with keyboard or random?(1 | 2)\n-->"**);**

scanf**(**"%i"**,&**k**);**

**if(**k**==**1**){**

inputKey**(**a**,**n**,**0**,**'a'**);**

inputKey**(**b**,**n**,**0**,**'b'**);**

**}**

**else{**

inputRand**(**a**,**n**,**0**);**

inputRand**(**b**,**n**,**0**);**

**}**

printf**(**"Array A:\n"**);**

out**(**a**,**n**,**0**);**

printf**(**"\nArray B:\n"**);**

out**(**b**,**n**,**0**);**

printf**(**"\nAnswer: %f"**,**sum1**(**a**,**b**,**n**,**0**));**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

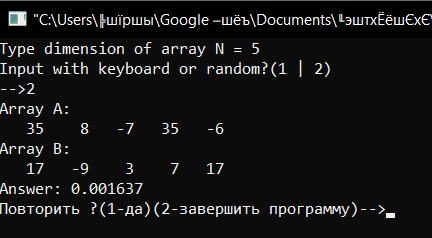
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Задание 6.**

**Условие:**

## 

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

#include <time.h>

#include <math.h>

#include <graphics.h>

void rec**(**int x**,**int y**,**int n**){**

**if(**x**>=**n**){**

**return;**

**}**

/\*

line(x,y,x+60,y);

line(x+60,y,x,y+60);

\*/

**for(**int i**=**x**;**i**<**x**+**60**;**i**++){**

putpixel**(**i**,**y**,**WHITE**);**

**}**

**for(**int i**=**y**;**i**<**y**+**60**;**i**++){**

putpixel**(**x**+**60**,**i**,**WHITE**);**

**}**

**return** rec**(**x**+**60**,**y**+**60**,**n**);**

**}**

int main**(){**

int ga**,**gm**;**

ga**=**DETECT**;**

gm**=**DETECT**;**

initgraph**(&**ga**,&**gm**,**""**);**

setcolor**(**3**);**

setlinestyle**(**0**,**1**,**1**);**

int i**,**j**;**

int n **=** 200**;**

int x**=**0**,**y**=**0**;**

setfillstyle**(**6**,**1**);**

rec**(**0**,**15 **,**n**);**

**for(**i**=**0**;**i**<**n**+**40**;**i**++){**

putpixel**(**i**,**255**,**WHITE**);**

**}**

**for(**i**=**0**;**i**<**n**+**55**;**i**++){**

putpixel**(**1**,**i**,**WHITE**);**

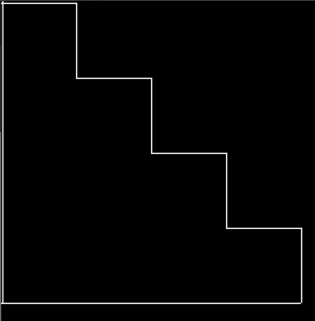
**}**

getch**();**

closegraph**();**

**}**

**Консоль:**

****

**Лабораторная работа №17.**

**Задание 1.**

**Условие:**

## Выполнить задание из предыдущих лабораторных работ по обработке строк, используя текстовые файлы.

**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <color.h>

#include <iostream>

#define forI for(int i=0;str[i]!='\0';i++)

#define forJ for(int j=0;j < n;j++)

void dell**(**char **\***s**,**char **\***s1**,**int i1**,** int j1**,**int length**){**

int dif**=**j1**-**i1**+**1**;**

length**-=**dif**;**

**for(**int i**=**i1**;**s**[**i**+**dif**]!=**'\0'**;**i**++){**

s1**[**i**]** **=** s**[**i**+**dif**];**

**}**

s1**[**length**]** **=** '\0'**;**

**}**

int main**(){**

FILE **\***f**;**

char str1**[**100**],**str**[**100**];**

int len**;**

f**=**fopen**(**"1.txt"**,**"w+"**);**

gets**(**str**);**

fprintf**(**f**,**"%s\n"**,**str**);**

printf**(**"\nThe file contains this string:\n"**);**

rewind**(**f**);**

**while(!**feof**(**f**)){**

fputc**(**fgetc**(**f**),**stdout**);**

**}**

len **=** strlen**(**str**);**

forI**{**

str1**[**i**]** **=** str**[**i**];**

**}**

str1**[**len**]=**'\0'**;**

forI**{**

**if(**str**[**i**]==**'"'**){**

**for(**int j**=**i**+**1**;**str**[**j**]!=**'\0'**;**j**++){**

**if(**str**[**j**]==**'"'**){**

dell**(**str**,**str1**,**i**,**j**,**len**);**

**}**

**}**

**}**

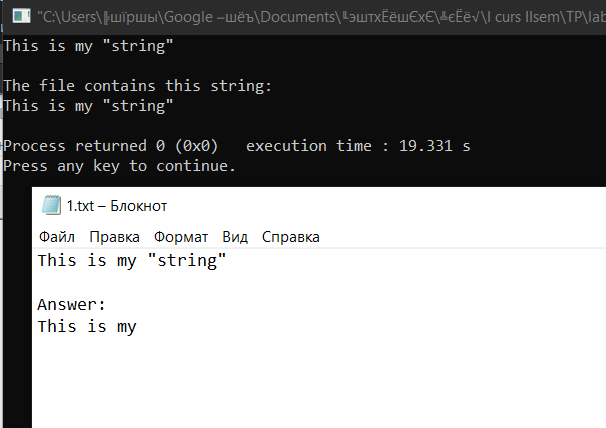
**}**

fprintf**(**f**,**"\nAnswer:\n"**);**

fprintf**(**f**,**"%s"**,**str1**);**

**}**

**Консоль:**



**Задание 2.**

**Условие:**

## Выполнить задание по одномерному массиву используя текстовые файлы.

**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <windows.h>

#define forLoop for(int i=0;i < n;i++)

bool del**(**int x**,**int y**){**

**if(**x**%**y**==**0**){**

**return** **true;**

**}**

**else{**

**return** **false;**

**}**

**}**

int Max**(**int o**,**int y**){**

**if(**o**>**y**){**

**return** o**;**

**}**

**else** **return** y**;**

**}**

int Min**(**int o**,**int y**){**

**if(**o**>**y**){**

**return** y**;**

**}**

**else** **return** o**;**

**}**

void outputArr**(**int **\***x**,**int n**){**

printf**(**"\nArray:\n"**);**

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

printf**(**"%6i"**,**x**[**i**]);**

**}**

**}**

void inputArrRand**(**int **\***x**,**int n**){**

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

x**[**i**]** **=** rand**()%**100**-**20**;**

**}**

**}**

void inputArrKeyB**(**int **\***x**,**int n**){**

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

printf**(**"A(%i) = "**,**i**);**

scanf**(**"%i"**,&**x**[**i**]);**

**}**

**}**

void f1**(**int x**[**20**],**int n**,**FILE **\***f**){**

int k**,**min**=**999999**,**max**=-**999999**;**

printf**(**"\nK = "**);**

scanf**(**"%i"**,&**k**);**

forLoop**{**

**if((**2**\***i**+**1**+**k**)>=**n**)** **break;**

**else**

min **=** Min**(**x**[**2**\***i**+**1**+**k**],**min**);**

**}**

forLoop**{**

i**++;**

**if((**2**\***i**+**1**+**k**)>=**n**)** **break;**

**else**

max **=** Max**(**x**[**2**\***i**+**1**],**max**);**

**}**

printf**(**"Max(%i,%i) = %i"**,**min**,**max**,**Max**(**min**,**max**));**

fprintf**(**f**,**"Max(%i,%i) = %i"**,**min**,**max**,**Max**(**min**,**max**));**

**}**

void f2**(**int x**[**20**],**int n**,**FILE **\***f**){**

int sum**=**0**,**pr**=**1**;**

forLoop**{**

**if(**del**(**x**[**i**],**3**))**sum**+=**x**[**i**];**

**if(**del**(**x**[**i**],**5**)&&**x**[**i**]!=**0**)**pr**\*=**x**[**i**];**

**}**

printf**(**"\nMax(%i,%i) = %i"**,**sum**,**pr**,**Max**(**sum**,**pr**));**

fprintf**(**f**,**"\nMax(%i,%i) = %i"**,**sum**,**pr**,**Max**(**sum**,**pr**));**

**}**

void f3**(**int x**[**20**],**int n**,**FILE **\***f**){**

printf**(**"\nРешение\n"**);**

forLoop**{**

**if(**x**[**i**]==**0**){**

x**[**i**]=**5**;**

**}**

**}**

outputArr**(**x**,**n**);**

fprintf**(**f**,**"\nРешение:\n"**);**

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

fprintf**(**f**,**"%5i"**,**x**[**i**]);**

**}**

**}**

void f4**(**int x**[**20**],**int n**,**FILE **\***f**){**

int pr**=**1**,**sum**=**0**,**k**=**0**;**

int p**,**t**;**

printf**(**"\np = "**);**

scanf**(**"%i"**,&**p**);**

printf**(**"t = "**);**

scanf**(**"%i"**,&**t**);**

forLoop**{**

**if(**x**[**i**]<**0**){**

pr**\*=**x**[**i**];**

**}**

i**++;**

**}**

**if(**pr**>**0**){**

forLoop**{**

**if(**del**(**x**[**i**],**p**))**sum**+=**x**[**i**];**

**}**

**if(**sum**<**t**){**

fprintf**(**f**,**"\nСумма элементов, делящихся на %i (%i) меньше %i"**,**p**,**sum**,**t**);**

printf**(**"\nСумма элементов, делящихся на %i (%i) меньше %i"**,**p**,**sum**,**t**);**

**}**

**else{**

fprintf**(**f**,**"\nСумма элементов, делящихся на %i (%i) не меньше %i"**,**p**,**sum**,**t**);**

printf**(**"\nСумма элементов, делящихся на %i (%i) не меньше %i"**,**p**,**sum**,**t**);**

**}**

**}**

**else{**

fprintf**(**f**,**"\nПроизведение отрицательных меньше нуля!!"**);**

printf**(**"\nПроизведение отрицательных меньше нуля!!"**);**

**}**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

Menu**:**

int k**,**N**;**

char ch**;**

printf**(**"\nВведите размерность массива N = "**);**

scanf**(**"%i"**,&**N**);**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

int a**[**N**];**

**switch(**k**){**

**case** 1**:**

inputArrKeyB**(**a**,**N**);**

**break;**

**case** 2**:**

inputArrRand**(**a**,**N**);**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

outputArr**(**a**,**N**);**

FILE **\***f**;**

f **=** fopen**(**"2.txt"**,**"w+"**);**

fprintf**(**f**,**"Array A\n"**);**

**for(**int i**=**0**;**i**<**N**;**i**++){**

fprintf**(**f**,**"%6i"**,**a**[**i**]);**

**}**

M1**:**

printf**(**"\nВыберите случай:\n1)max(min(a[1+k],a[3+k],..),max(нечет))\n2)max(summ(эл-ов делящ. на 3), произведения элементов делящихся на 5)\n3)все эл-ты - 0 заменить на 5\n4)если произведение(отрицательных на четных поз)>0, определить если сумма элементов делящихся на p < t\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

f1**(**a**,**N**,**f**);**

**break;**

**case** '2'**:**

f2**(**a**,**N**,**f**);**

**break;**

**case** '3'**:**

f3**(**a**,**N**,**f**);**

**break;**

**case** '4'**:**

f4**(**a**,**N**,**f**);**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

fclose**(**f**);**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** Menu**;**

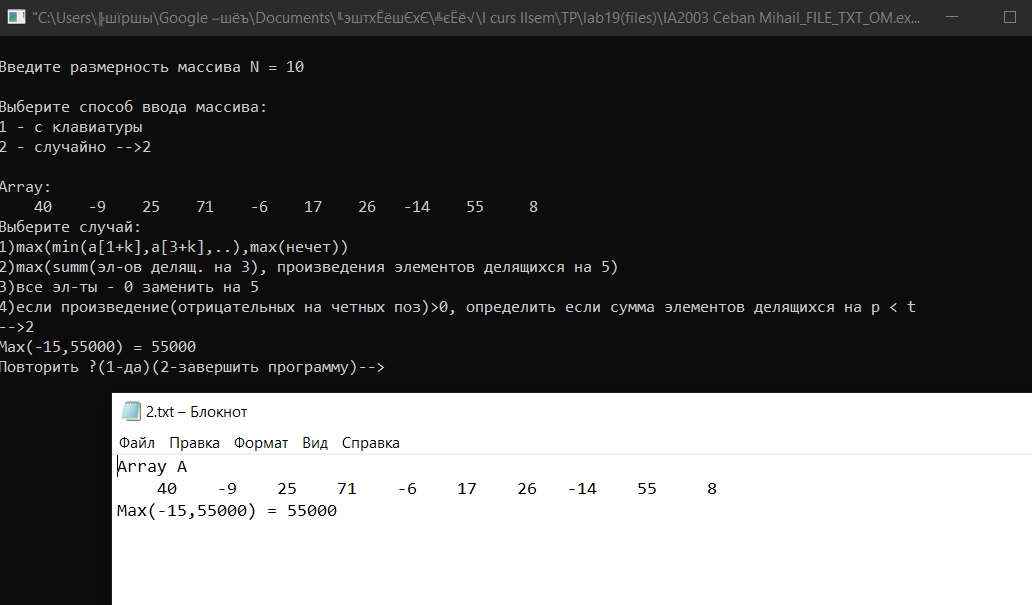
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**



**Задание 3.**

**Условие:**

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

struct address**{**

char city**[**20**];**

char street**[**20**];**

int nr**;**

**};**

struct delivery**{**

struct address addr**;**

char tel**[**15**];**

char payment**[**10**];**

**};**

struct charact**{**

int length**;**

int width**;**

int height**;**

**};**

struct tip**{**

char name**[**20**];**

struct charact har**;**

float price**;**

**};**

struct mebel**{**

struct tip type**;**

char manufacturer**[**20**];**

int warranty**;**

struct delivery del**;**

**};**

void input**(**struct mebel x**[],**int m**,**int n**){**

bool b**=false;**

**for(**int i**=**m**;**i**<**n**;**i**++){**

printf**(**"\nOrder nr. %i\n"**,**i**+**1**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of furniture:"**);**scanf**(**"%s"**,**x**[**i**].**type**.**name**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**type**.**name**);**j**++){**

b **=** isalpha**(**x**[**i**].**type**.**name**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Length of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**length**);**

printf**(**"Width of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**width**);**

printf**(**"Height of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**height**);**

printf**(**"Price of furniture(in dollars):"**);**scanf**(**"%f"**,&**x**[**i**].**type**.**price**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of manufacturer:"**);**scanf**(**"%s"**,**x**[**i**].**manufacturer**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**manufacturer**);**j**++){**

b **=** isalpha**(**x**[**i**].**manufacturer**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Telephone number:"**);**scanf**(**"%s"**,**x**[**i**].**del**.**tel**);**

printf**(**"Warranty(months):"**);**scanf**(**"%i"**,&**x**[**i**].**warranty**);**

b**=false;**

**while(!**b**){**

printf**(**"Payment method(card or cash):"**);**scanf**(**"%s"**,**x**[**i**].**del**.**payment**);**

**if(!**strcmp**(**x**[**i**].**del**.**payment**,**"card"**)||!**strcmp**(**x**[**i**].**del**.**payment**,**"cash"**)){**

b **=** **true;**

**}**

**else{**

printf**(**"You need to enter words cash or card!\n"**);**

**}**

**}**

printf**(**"Address of delivery:\n"**);**

printf**(**"City:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**city**);**

printf**(**"Street:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**street**);**

printf**(**"Nr. of house:"**);** scanf**(**"%i"**,&**x**[**i**].**del**.**addr**.**nr**);**

**}**

**}**

void outputFile**(**struct mebel x**[],**int m**,** int n**,**FILE **\***f**){**

fprintf**(**f**,**"-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

SetColor**(**3**);**

fprintf**(**f**,**"\n| Name | Length | Width | Height | Price | Manufacturer | Warranty | Tel. number | Payment | City | Street |Nr. of house|"**);**

SetColor**(**15**);**

fprintf**(**f**,**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**for** **(**int i **=** m**;** i **<** n**;** i**++)**

**{**

fprintf**(**f**,**"\n|%12s|%8i sm.|%7i sm.|%8i sm.|%10.2f$|%16s|%7i months|%15s|%9s|%10s|%14s|%12i|"**,**x**[**i**].**type**.**name**,**x**[**i**].**type**.**har**.**length**,**x**[**i**].**type**.**har**.**width**,**x**[**i**].**type**.**har**.**height**,**

x**[**i**].**type**.**price**,**x**[**i**].**manufacturer**,**x**[**i**].**warranty**,**x**[**i**].**del**.**tel**,**x**[**i**].**del**.**payment**,**x**[**i**].**del**.**addr**.**city**,**x**[**i**].**del**.**addr**.**street**,**x**[**i**].**del**.**addr**.**nr**);**

fprintf**(**f**,**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**}**

**}**

void output**(**struct mebel x**[],**int m**,** int n**){**

printf**(**"-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

SetColor**(**3**);**

printf**(**"\n| Name | Length | Width | Height | Price | Manufacturer | Warranty | Tel. number | Payment | City | Street |Nr. of house|"**);**

SetColor**(**15**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**for** **(**int i **=** m**;** i **<** n**;** i**++)**

**{**

printf**(**"\n|%12s|%8i sm.|%7i sm.|%8i sm.|%10.2f$|%16s|%7i months|%15s|%9s|%10s|%14s|%12i|"**,**x**[**i**].**type**.**name**,**x**[**i**].**type**.**har**.**length**,**x**[**i**].**type**.**har**.**width**,**x**[**i**].**type**.**har**.**height**,**

x**[**i**].**type**.**price**,**x**[**i**].**manufacturer**,**x**[**i**].**warranty**,**x**[**i**].**del**.**tel**,**x**[**i**].**del**.**payment**,**x**[**i**].**del**.**addr**.**city**,**x**[**i**].**del**.**addr**.**street**,**x**[**i**].**del**.**addr**.**nr**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**}**

**}**

void findEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

bool b**=false;**

printf**(**"\nEnter the name of furniture:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

output**(**x**,**i**,**i**+**1**);**

b **=** **true;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**}**

void delEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

int index**=**0**;**

bool b**=false;**

printf**(**"\nEnter the name of furniture, that you want to delete:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

b **=** **true;**

index **=** i**;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**else{**

**for(**int i**=**index**;**i**<**n**-**1**;**i**++){**

x**[**i**]** **=** x**[**i**+**1**];**

**}**

**}**

**}**

int main**(){**

SetColor**(**15**);**

int n**=**2**,**e**=**3**,**m**=**0**;**

printf**(**"Type number of orders:"**);**scanf**(**"%i"**,&**n**);**

struct mebel a**[**10**];**

struct mebel b**[**10**];**

struct mebel temp**;**

int k**,**menuCase**;**

FILE **\***f**;**

f **=** fopen**(**"struct.txt"**,**"w+"**);**

M1**:**

printf**(**"1)Example\n2)Input struct.\n-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:**

**for(**int i**=**0**;**i**<**e**;**i**++){**

b**[**i**].**type**.**name **=** **{**'C'**,**'h'**,**'a'**,**'i'**,**'r'**};**

b**[**i**].**type**.**har**.**length **=** 100**;**

b**[**i**].**type**.**har**.**height **=** 150**;**

b**[**i**].**type**.**har**.**width **=** 50**;**

b**[**i**].**type**.**price **=** 15.25**;**

b**[**i**].**manufacturer **=** **{**'J'**,**'y'**,**'s'**,**'k'**};**

b**[**i**].**warranty **=** 13**;**

b**[**i**].**del**.**addr**.**city **=** **{**'C'**,**'h'**,**'i'**,**'s'**,**'i'**,**'n'**,**'a'**,**'u'**};**

b**[**i**].**del**.**addr**.**street **=** **{**'D'**,**'a'**,**'c'**,**'i'**,**'a'**};**

b**[**i**].**del**.**addr**.**nr **=** 24**;**

b**[**i**].**del**.**payment **=** **{**'c'**,**'a'**,**'s'**,**'h'**};**

b**[**i**].**del**.**tel **=** **{**'0'**,**'2'**,**'2'**,**'3'**,**'4'**,**'5'**,**'6'**,**'7'**};**

**}**

b**[**1**].**type**.**name **=** **{**'D'**,**'e'**,**'s'**,**'k'**};**

b**[**2**].**type**.**name **=** **{**'B'**,**'e'**,**'d'**};**

b**[**1**].**type**.**price **=** 30**;**

b**[**2**].**type**.**price **=** 50**;**

**break;**

**case** 2**:**

input**(**a**,**m**,**n**);**

**break;**

**default:**

printf**(**"Wrong value! Try again!\n"**);**

**goto** M1**;**

**}**

fprintf**(**f**,**"\nSTRUCT BEFORE\n"**);**

**if(**k**==**1**){**

outputFile**(**b**,**m**,**e**,**f**);**

**}**

**else{**

outputFile**(**a**,**m**,**n**,**f**);**

**}**

Menu**:**

printf**(**"\nMenu:\n0)Exit\n1)Output struct\n2)Find element by the name\n3)Insert one element to struct\n4)Delete an element in the struct\n5)Sort by price\n-->"**);**

scanf**(**"%i"**,&**menuCase**);**

**switch(**menuCase**){**

**case** 0**:**

**goto** End**;**

**break;**

**case** 1**:**

**if(**k**==**1**){**

output**(**b**,**m**,**e**);**

**}**

**else{**

output**(**a**,**m**,**n**);**

**}**

**break;**

**case** 2**:**

**if(**k**==**1**){**

findEl**(**b**,**e**);**

**}**

**else{**

findEl**(**a**,**n**);**

**}**

**break;**

**case** 3**:**

**if(**k**==**1**){**

input**(**b**,**e**,**e**+**1**);**

e**++;**

**}**

**else{**

input**(**a**,**n**,**n**+**1**);**

n**++;**

**}**

**break;**

**case** 4**:**

**if(**k**==**1**){**

delEl**(**b**,**e**);**

e**--;**

**}**

**else{**

delEl**(**a**,**n**);**

n**--;**

**}**

**break;**

**case** 5**:**

**if(**k**==**1**){**

**for(**int i**=**0**;**i**<**e**-**1**;**i**++){**

**for(**int j**=**0**;**j**<**e**-**1**-**i**;**j**++){**

**if(**b**[**j**].**type**.**price**<**b**[**j**+**1**].**type**.**price**){**

temp **=** b**[**j**];**

b**[**j**]** **=** b**[**j**+**1**];**

b**[**j**+**1**]** **=** temp**;**

**}**

**}**

**}**

**}**

**else{**

**for(**int i**=**0**;**i**<**n**-**1**;**i**++){**

**for(**int j**=**0**;**j**<**n**-**1**-**i**;**j**++){**

**if(**a**[**j**].**type**.**price**<**a**[**j**+**1**].**type**.**price**){**

temp **=** a**[**j**];**

a**[**j**]** **=** a**[**j**+**1**];**

a**[**j**+**1**]** **=** temp**;**

**}**

**}**

**}**

**}**

**break;**

**}**

**goto** Menu**;**

End**:**

fprintf**(**f**,**"\nSTRUCT AFTER\n"**);**

**if(**k**==**1**){**

outputFile**(**b**,**m**,**e**,**f**);**

**}**

**else{**

outputFile**(**a**,**m**,**n**,**f**);**

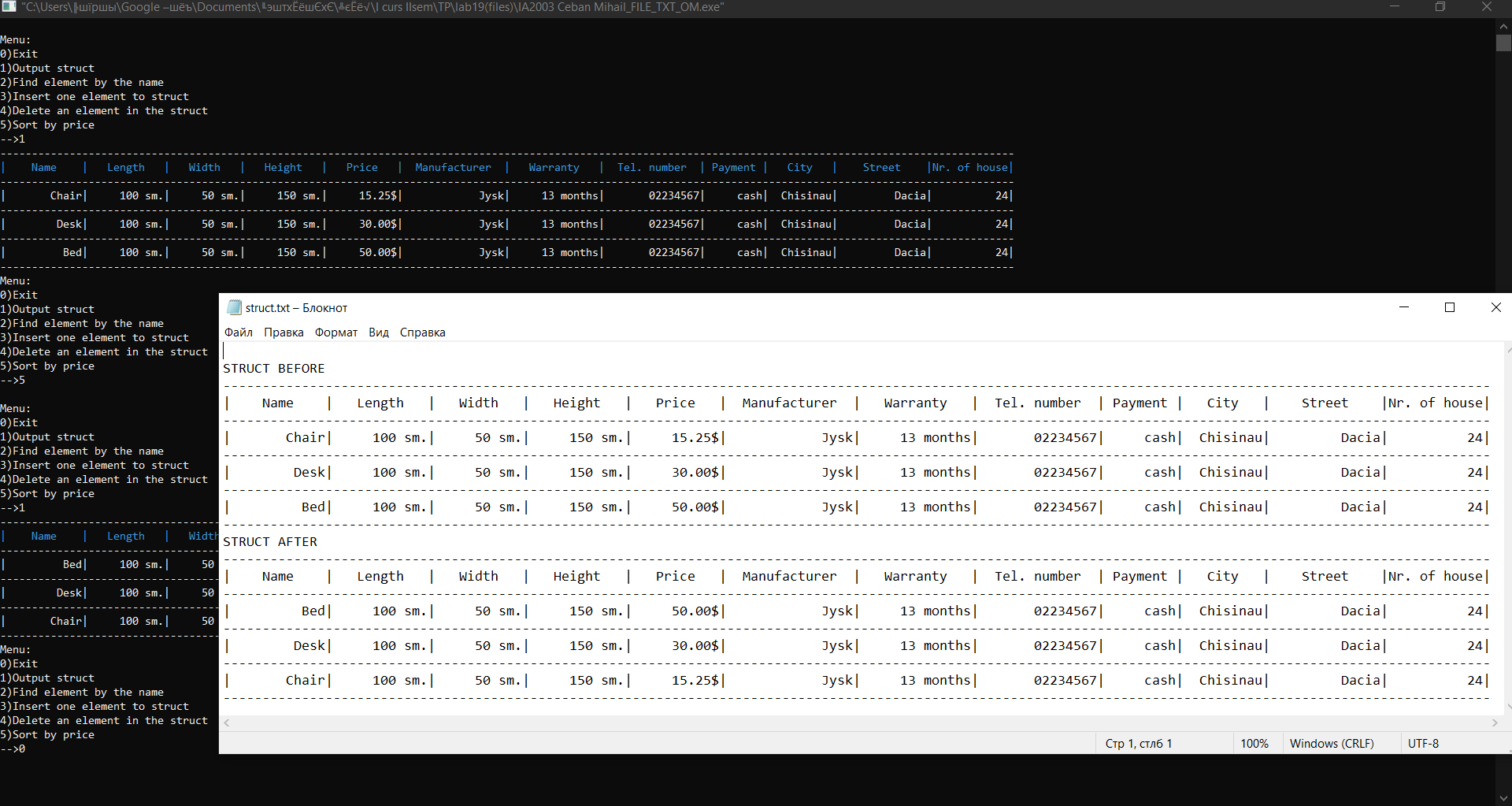
**}**

fclose**(**f**);**

getche**();**

**}**

**Консоль:**



**Лабораторная работа №18.**

**Задание 1.**

**Условие:**

## Выполнить задание по одномерному массиву используя Бинарные файлы.

**Программа:**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main(){

int n, i;

float a;

FILE \*f;

f = fopen("mih.dat", "wb");

printf("n=");

scanf("%i",&n);

fwrite(&n, sizeof(int), 1, f);

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

printf("element %i = ",i+1);

scanf("%f",&a);

//printf("%.")

fwrite(&a,sizeof(float), 1, f);

}

fclose(f);

n = 0;

f = fopen("mih.dat", "rb+");

fread(&n, sizeof(int), 1, f);

printf("n = %i\n",n);

float aa[n];

fread(aa, sizeof(float), n, f);

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

printf("%7.2f",aa[i]);

}

printf("\n");

int k,poz,quant;

float sum=0,elem;

float med=0;

printf("\n1)SUM of positive elements\n2)Quantity of negative elements\n3)Media of all elements\n-->");

scanf("%i",&k);

switch(k){

case 1:

sum=0;

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

if(aa[i]>0){

poz = i;

fseek(f,sizeof(int)+poz\*sizeof(float), 0);

fread(&elem,sizeof(float), 1, f);

printf("\n%f",elem);

sum+=elem;

}

}

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fwrite(&sum,sizeof(float), 1, f);

sum = 0;

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fread(&sum,sizeof(float), 1, f);

printf("\nAnswer is %.2f",sum);

break;

case 2:

quant = 0;

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

if(aa[i]<0){

quant++;

}

}

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fwrite(&quant,sizeof(int), 1, f);

quant = 0;

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fread(&quant,sizeof(int), 1, f);

printf("\nAnswer is %i",quant);

break;

case 3:

quant = 0;

sum=0;

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

poz = i;

fseek(f,sizeof(int)+poz\*sizeof(float), 0);

fread(&elem,sizeof(float), 1, f);

sum+=elem;

quant++;

}

med = sum/quant;

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fwrite(&med,sizeof(float), 1, f);

med = 0;

fseek(f,sizeof(int)+n\*sizeof(float), 0);

fread(&med,sizeof(float), 1, f);

printf("\nAnswer is %.2f",med);

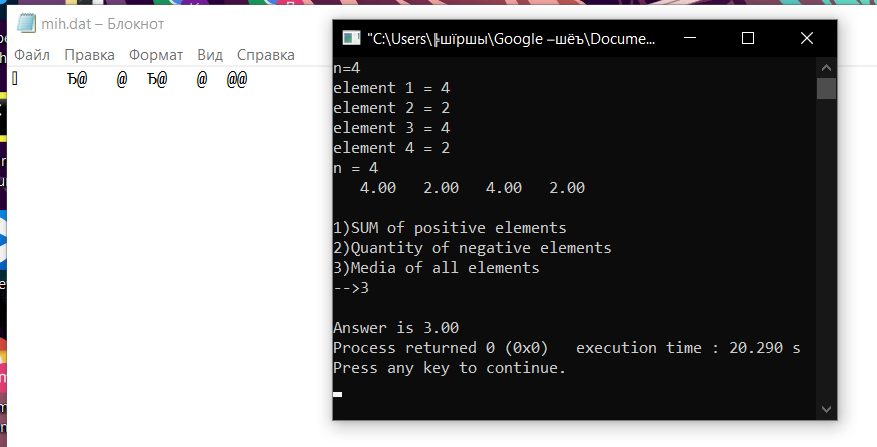
break;

}

return 0;

}

**Консоль:**

****

**Задание 2.**

**Условие:**

## Выполнить задание со структурой используя бинарные файлы

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

FILE **\***f**;**

struct address**{**

char city**[**20**];**

char street**[**20**];**

int nr**;**

**};**

struct delivery**{**

struct address addr**;**

char tel**[**15**];**

char payment**[**10**];**

**};**

struct charact**{**

int length**;**

int width**;**

int height**;**

**};**

struct tip**{**

char name**[**20**];**

struct charact har**;**

float price**;**

**};**

struct mebel**{**

struct tip type**;**

char manufacturer**[**20**];**

int warranty**;**

struct delivery del**;**

**};**

void input**(**struct mebel x**[],**int m**,**int n**){**

bool b**=false;**

**for(**int i**=**m**;**i**<**n**;**i**++){**

printf**(**"\nOrder nr. %i\n"**,**i**+**1**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of furniture:"**);**scanf**(**"%s"**,**x**[**i**].**type**.**name**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**type**.**name**);**j**++){**

b **=** isalpha**(**x**[**i**].**type**.**name**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Length of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**length**);**

printf**(**"Width of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**width**);**

printf**(**"Height of furniture(in cm):"**);**scanf**(**"%i"**,&**x**[**i**].**type**.**har**.**height**);**

printf**(**"Price of furniture(in dollars):"**);**scanf**(**"%f"**,&**x**[**i**].**type**.**price**);**

b**=false;**

**while(!**b**){**

printf**(**"Name of manufacturer:"**);**scanf**(**"%s"**,**x**[**i**].**manufacturer**);**

**for(**int j**=**0**;**j**<**strlen**(**x**[**i**].**manufacturer**);**j**++){**

b **=** isalpha**(**x**[**i**].**manufacturer**[**j**]);**

**if(!**b**){**

printf**(**"Wrong name!Try again!Only letters required!\n"**);**

**break;**

**}**

**}**

**}**

printf**(**"Telephone number:"**);**scanf**(**"%s"**,**x**[**i**].**del**.**tel**);**

printf**(**"Warranty(months):"**);**scanf**(**"%i"**,&**x**[**i**].**warranty**);**

b**=false;**

**while(!**b**){**

printf**(**"Payment method(card or cash):"**);**scanf**(**"%s"**,**x**[**i**].**del**.**payment**);**

**if(!**strcmp**(**x**[**i**].**del**.**payment**,**"card"**)||!**strcmp**(**x**[**i**].**del**.**payment**,**"cash"**)){**

b **=** **true;**

**}**

**else{**

printf**(**"You need to enter words cash or card!\n"**);**

**}**

**}**

printf**(**"Address of delivery:\n"**);**

printf**(**"City:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**city**);**

printf**(**"Street:"**);** scanf**(**"%s"**,**x**[**i**].**del**.**addr**.**street**);**

printf**(**"Nr. of house:"**);** scanf**(**"%i"**,&**x**[**i**].**del**.**addr**.**nr**);**

**}**

**}**

void binarCoding**(**struct mebel x**[],**int n**){**

f **=** fopen**(**"struct.dat"**,**"wb"**);**

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

fwrite**(&**x**[**i**],sizeof(**x**[**i**]),**1**,**f**);**

**}**

fclose**(**f**);**

**}**

void getStuct**(**struct mebel x**[],**int n**){**

f **=** fopen**(**"struct.dat"**,**"rb"**);**

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

fread**(&**x**[**i**],sizeof(**x**[**i**]),**1**,**f**);**

**}**

fclose**(**f**);**

**}**

void output**(**struct mebel x**[],**int m**,** int n**){**

printf**(**"-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

SetColor**(**3**);**

printf**(**"\n| Name | Length | Width | Height | Price | Manufacturer | Warranty | Tel. number | Payment | City | Street |Nr. of house|"**);**

SetColor**(**15**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**for** **(**int i **=** m**;** i **<** n**;** i**++)**

**{**

printf**(**"\n|%12s|%8i sm.|%7i sm.|%8i sm.|%10.2f$|%16s|%7i months|%15s|%9s|%10s|%14s|%12i|"**,**x**[**i**].**type**.**name**,**x**[**i**].**type**.**har**.**length**,**x**[**i**].**type**.**har**.**width**,**x**[**i**].**type**.**har**.**height**,**

x**[**i**].**type**.**price**,**x**[**i**].**manufacturer**,**x**[**i**].**warranty**,**x**[**i**].**del**.**tel**,**x**[**i**].**del**.**payment**,**x**[**i**].**del**.**addr**.**city**,**x**[**i**].**del**.**addr**.**street**,**x**[**i**].**del**.**addr**.**nr**);**

printf**(**"\n-----------------------------------------------------------------------------------------------------------------------------------------------------------------"**);**

**}**

**}**

void findEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

bool b**=false;**

printf**(**"\nEnter the name of furniture:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

output**(**x**,**i**,**i**+**1**);**

b **=** **true;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**}**

void delEl**(**struct mebel x**[],**int n**){**

char s**[**20**];**

int index**=**0**;**

bool b**=false;**

printf**(**"\nEnter the name of furniture, that you want to delete:"**);**

scanf**(**"%s"**,&**s**);**

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

**if(!**strcmp**(**x**[**i**].**type**.**name**,**s**)){**

b **=** **true;**

index **=** i**;**

**break;**

**}**

**}**

**if(!**b**){**

printf**(**"The furniture with such name doesn't exist!!!"**);**

**}**

**else{**

**for(**int i**=**index**;**i**<**n**-**1**;**i**++){**

x**[**i**]** **=** x**[**i**+**1**];**

**}**

**}**

**}**

int main**(){**

SetColor**(**15**);**

int n**=**2**,**e**=**3**,**m**=**0**;**

printf**(**"Type number of orders:"**);**

scanf**(**"%i"**,&**n**);**

struct mebel a**[**10**];**

struct mebel b**[**10**];**

int k**,**menuCase**;**

struct mebel temp**;**

int e1**=**e**,**n1**=**n**;**

M1**:**

printf**(**"1)Example\n2)Input struct.\n-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:**

**for(**int i**=**0**;**i**<**e**;**i**++){**

b**[**i**].**type**.**name **=** **{**'C'**,**'h'**,**'a'**,**'i'**,**'r'**};**

b**[**i**].**type**.**har**.**length **=** 100**;**

b**[**i**].**type**.**har**.**height **=** 150**;**

b**[**i**].**type**.**har**.**width **=** 50**;**

b**[**i**].**type**.**price **=** 15.25**;**

b**[**i**].**manufacturer **=** **{**'J'**,**'y'**,**'s'**,**'k'**};**

b**[**i**].**warranty **=** 13**;**

b**[**i**].**del**.**addr**.**city **=** **{**'C'**,**'h'**,**'i'**,**'s'**,**'i'**,**'n'**,**'a'**,**'u'**};**

b**[**i**].**del**.**addr**.**street **=** **{**'D'**,**'a'**,**'c'**,**'i'**,**'a'**};**

b**[**i**].**del**.**addr**.**nr **=** 24**;**

b**[**i**].**del**.**payment **=** **{**'c'**,**'a'**,**'s'**,**'h'**};**

b**[**i**].**del**.**tel **=** **{**'0'**,**'2'**,**'2'**,**'3'**,**'4'**,**'5'**,**'6'**,**'7'**};**

**}**

b**[**1**].**type**.**name **=** **{**'D'**,**'e'**,**'s'**,**'k'**};**

b**[**2**].**type**.**name **=** **{**'B'**,**'e'**,**'d'**};**

binarCoding**(**b**,**e**);**

**break;**

**case** 2**:**

input**(**a**,**m**,**n**);**

binarCoding**(**a**,**n**);**

**break;**

**default:**

printf**(**"Wrong value! Try again!\n"**);**

**goto** M1**;**

**}**

printf**(**"Saving the stucture in binare file..."**);**

Menu**:**

printf**(**"\nMenu:\n0)Exit\n1)Output struct\n2)Find element by the name\n3)Insert one element to struct\n4)Delete an element in the struct\n5)Sort by price\n6)Get saved struct from binare file\n-->"**);**

scanf**(**"%i"**,&**menuCase**);**

**switch(**menuCase**){**

**case** 0**:**

**goto** End**;**

**break;**

**case** 1**:**

**if(**k**==**1**){**

output**(**b**,**m**,**e**);**

**}**

**else{**

output**(**a**,**m**,**n**);**

**}**

**break;**

**case** 2**:**

**if(**k**==**1**){**

findEl**(**b**,**e**);**

**}**

**else{**

findEl**(**a**,**n**);**

**}**

**break;**

**case** 3**:**

**if(**k**==**1**){**

input**(**b**,**e**,**e**+**1**);**

e**++;**

**}**

**else{**

input**(**a**,**n**,**n**+**1**);**

n**++;**

**}**

**break;**

**case** 4**:**

**if(**k**==**1**){**

delEl**(**b**,**e**);**

e**--;**

**}**

**else{**

delEl**(**a**,**n**);**

n**--;**

**}**

**break;**

**case** 5**:**

**if(**k**==**1**){**

**for(**int i**=**0**;**i**<**e**-**1**;**i**++){**

**for(**int j**=**0**;**j**<**e**-**1**-**i**;**j**++){**

**if(**b**[**j**].**type**.**price**<**b**[**j**+**1**].**type**.**price**){**

temp **=** b**[**j**];**

b**[**j**]** **=** b**[**j**+**1**];**

b**[**j**+**1**]** **=** temp**;**

**}**

**}**

**}**

**}**

**else{**

**for(**int i**=**0**;**i**<**n**-**1**;**i**++){**

**for(**int j**=**0**;**j**<**n**-**1**-**i**;**j**++){**

**if(**a**[**j**].**type**.**price**<**a**[**j**+**1**].**type**.**price**){**

temp **=** a**[**j**];**

a**[**j**]** **=** a**[**j**+**1**];**

a**[**j**+**1**]** **=** temp**;**

**}**

**}**

**}**

**}**

printf**(**"\nSuccess!"**);**

**break;**

**case** 6**:**

**if(**k**==**1**){**

e**=**e1**;**

getStuct**(**b**,**e**);**

**}**

**else{**

n**=**n1**;**

getStuct**(**a**,**n**);**

**}**

printf**(**"\nSuccess!"**);**

**}**

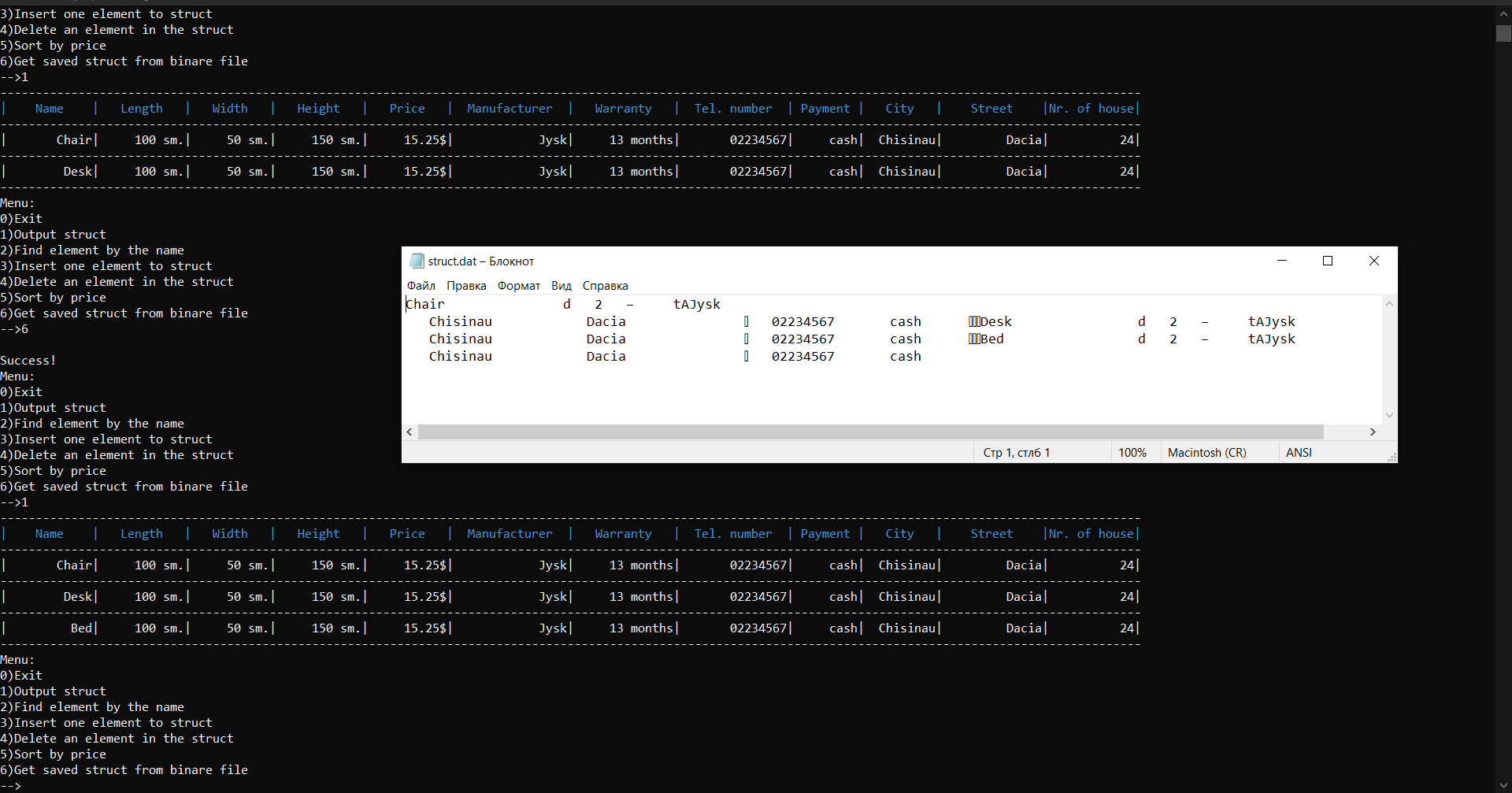
**goto** Menu**;**

End**:**

getche**();**

**}**

**Консоль:**

****

**Лабораторная работа №19.**

**Задание 1.**

**Условие:**

## 

**Программа:**

#include <stdio.h>

#include <math.h>

#include <stdbool.h>

#include <conio.h>

#include <stdlib.h>

#include <time.h>

#include <windows.h>

#define forLoop for(int i=0;i < n;i++)

struct mylist

**{**

int a**;**

mylist **\***next**;**

**};**

mylist **\***first**,** **\***last**;**

void create\_list**(**int i**,**char chr**)**

**{**

mylist **\***curent**;**

**if(!**first**){**

first **=** **new** mylist**;**

**if** **(**chr **==** 'k'**){**

printf**(**"\na[%i] = "**,**i**);**

scanf**(**"%i"**,&**first**->**a**);**

**}**

**else{**

first**->**a **=** rand**()%**100**-**20**;**

**}**

last **=** first**;**

**}**

**else{**

curent **=** **new** mylist**;**

**if** **(**chr **==** 'k'**){**

printf**(**"a[%i] = "**,**i**);**

scanf**(**"%i"**,&**curent**->**a**);**

**}**

**else{**

curent**->**a **=** rand**()%**100**-**20**;**

**}**

last**->**next **=** curent**;**

last **=** curent**;**

**}**

last**->**next **=** 0**;**

**}**

bool del**(**int x**,**int y**){**

**if(**x**%**y**==**0**){**

**return** **true;**

**}**

**else{**

**return** **false;**

**}**

**}**

int Max**(**int o**,**int y**){**

**if(**o**>**y**){**

**return** o**;**

**}**

**else** **return** y**;**

**}**

int Min**(**int o**,**int y**){**

**if(**o**>**y**){**

**return** y**;**

**}**

**else** **return** o**;**

**}**

void outputArr**(){**

mylist **\***c**;**

c **=** first**;**

**while(**c**){**

printf**(**"%5i"**,**c**->**a**);**

c **=** c**->**next**;**

**}**

printf**(**"\n"**);**

**}**

void f1**(**int n**){**

int k**,**min**=**999999**,**max**=-**999999**;**

printf**(**"\nK = "**);**

scanf**(**"%i"**,&**k**);**

mylist **\***c**;**

c **=** first**;**

int p**;**

forLoop**{**

p **=** 2**\***i**+**1**+**k**;**

c **=** first**;**

**if(**p**>=**n**)** **break;**

**else{**

**for(**int j **=** 0**;**j**<**p**;**j**++){**

c **=** c**->**next**;**

**}**

min **=** Min**(**c**->**a**,**min**);**

**}**

**}**

forLoop**{**

i**++;**

p **=** 2**\***i**+**1**;**

c **=** first**;**

**if(**p**>=**n**)** **break;**

**else{**

p **=** 2**\***i**+**1**;**

**for(**int j **=** 0**;**j**<**p**;**j**++){**

c **=** c**->**next**;**

**}**

max **=** Max**(**c**->**a**,**max**);**

**}**

**}**

printf**(**"Max(%i,%i) = %i"**,**min**,**max**,**Max**(**min**,**max**));**

**}**

void f2**(**int n**){**

mylist **\***c**;**

c **=** first**;**

int sum**=**0**,**pr**=**1**;**

forLoop**{**

**if(**del**(**c**->**a**,**3**))**sum**+=**c**->**a**;**

**if(**del**(**c**->**a**,**5**)&&**c**->**a**!=**0**)**pr**\*=**c**->**a**;**

c **=** c**->**next**;**

**}**

printf**(**"\nMax(%i,%i) = %i"**,**sum**,**pr**,**Max**(**sum**,**pr**));**

**}**

void f3**(**int n**){**

mylist **\***c**;**

c **=** first**;**

printf**(**"\nРешение\n"**);**

forLoop**{**

**if(**c**->**a**==**0**){**

c**->**a**=**5**;**

**}**

c **=** c**->**next**;**

**}**

outputArr**();**

**}**

void f4**(**int n**){**

mylist **\***c**;**

c **=** first**;**

int pr**=**1**,**sum**=**0**,**k**=**0**;**

int p**,**t**;**

printf**(**"\np = "**);**

scanf**(**"%i"**,&**p**);**

printf**(**"t = "**);**

scanf**(**"%i"**,&**t**);**

forLoop**{**

**if(**c**->**a**<**0**){**

pr**\*=**c**->**a**;**

**}**

c **=** c**->**next**;**

**if(**i**+**1**<**n**-**1**){**

c **=** c**->**next**;**

i**++;**

**}**

**}**

c **=** first**;**

**if(**pr**>**0**){**

c **=** first**;**

forLoop**{**

**if(**del**(**c**->**a**,**p**))**sum**+=**c**->**a**;**

c **=** c**->**next**;**

**}**

**if(**sum**<**t**){**

printf**(**"\nСумма элементов, делящихся на %i (%i) меньше %i"**,**p**,**sum**,**t**);**

**}**

**else{**

printf**(**"\nСумма элементов, делящихся на %i (%i) не меньше %i"**,**p**,**sum**,**t**);**

**}**

**}**

**else{**

printf**(**"\nПроизведение отрицательных меньше нуля!!"**);**

**}**

**}**

int main**(){**

SetConsoleCP**(**1251**);**

SetConsoleOutputCP**(**1251**);**

srand**(**time**(NULL));**

int k**,**N**;**

char ch**;**

printf**(**"\nВведите размерность массива N = "**);**

scanf**(**"%i"**,&**N**);**

printf**(**"\nВыберите способ ввода массива:\n1 - с клавиатуры\n2 - случайно -->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:**

**for(**int i**=**0**;**i**<**N**;**i**++){**

create\_list**(**i**,**'k'**);**

**}**

**break;**

**case** 2**:**

**for(**int i**=**0**;**i**<**N**;**i**++){**

create\_list**(**i**,**'r'**);**

**}**

**break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

outputArr**();**

M1**:**

printf**(**"\nВыберите случай:\n1)max(min(a[1+k],a[3+k],..),max(нечет))\n2)max(summ(эл-ов делящ. на 3), произведения элементов делящихся на 5)\n3)все эл-ты - 0 заменить на 5\n4)если произведение(отрицательных на четных поз)>0, определить если сумма элементов делящихся на p < t\n-->"**);**

ch **=** getche**();**

**switch(**ch**){**

**case** '1'**:**

f1**(**N**);**

**break;**

**case** '2'**:**

f2**(**N**);**

**break;**

**case** '3'**:**

f3**(**N**);**

**break;**

**case** '4'**:**

f4**(**N**);**

**break;**

**default:**printf**(**"Недопустимое значение, попробуйте еще раз!!"**);goto** M1**;**

**}**

M3**:**

printf**(**"\nПовторить ?(1-да)(2-завершить программу)-->"**);**

scanf**(**"%i"**,&**k**);**

**switch(**k**){**

**case** 1**:goto** M1**;**

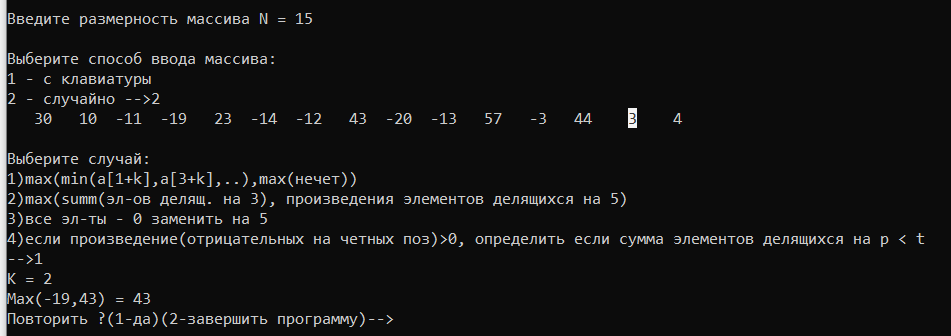
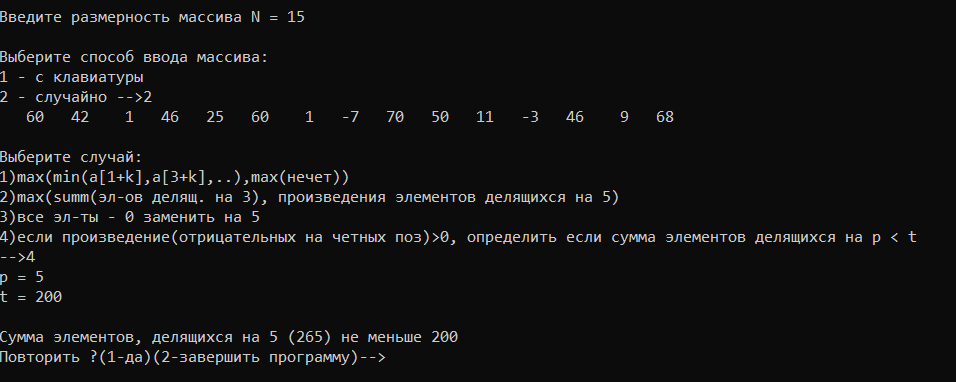
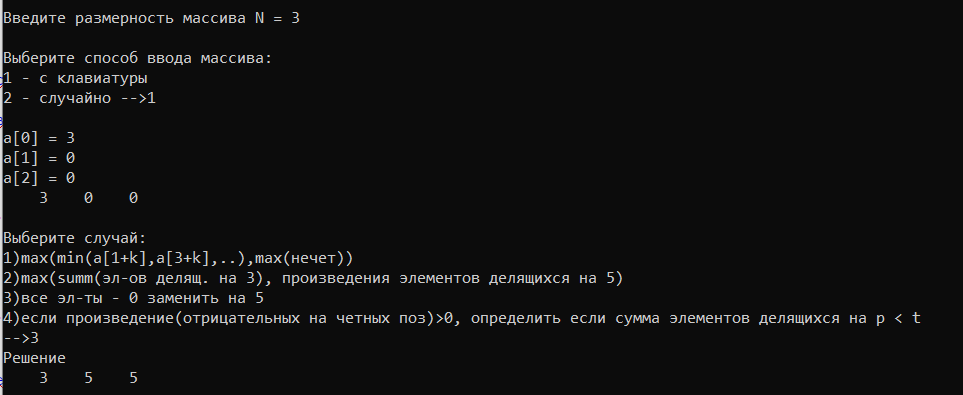
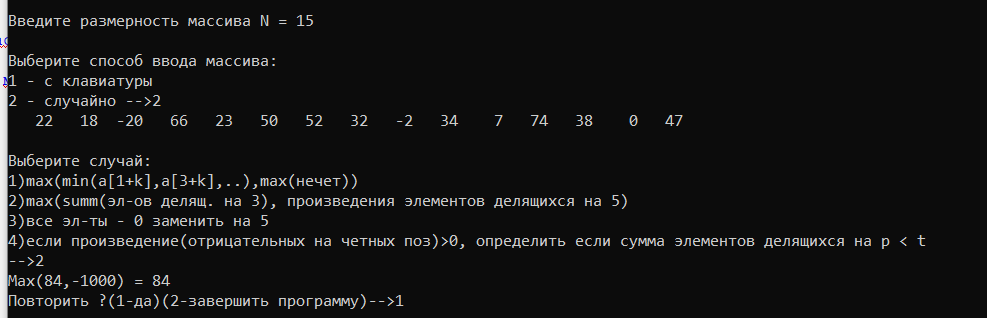
**case** 2**:break;**

**default:**printf**(**"Вы ввели неверное значение, попробуйте еще раз!!!"**);goto** M3**;**

**}**

**}**

**Консоль:**

****

**Задание 2.**

**Условие:**

****

**Программа:**

#include <color.h>

#include <conio.h>

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <ctype.h>

struct charact**{**

int length**;**

int width**;**

int height**;**

**};**

struct mylist

**{**

charact a**;**

mylist **\***next**;**

**};**

mylist **\***first**,** **\***last**;**

void create\_list**(**int i**)**

**{**

mylist **\***curent**;**

bool b**=false;**

**if(!**first**){**

first **=** **new** mylist**;**

printf**(**" length = "**);**

scanf**(**"%i"**,&**first**->**a**.**length**);**

printf**(**" width = "**);**

scanf**(**"%i"**,&**first**->**a**.**width**);**

printf**(**" height = "**);**

scanf**(**"%i"**,&**first**->**a**.**height**);**

last **=** first**;**

**}**

**else{**

curent **=** **new** mylist**;**

printf**(**" length = "**);**

scanf**(**"%i"**,&**curent**->**a**.**length**);**

printf**(**" width = "**);**

scanf**(**"%i"**,&**curent**->**a**.**width**);**

printf**(**" height = "**);**

scanf**(**"%i"**,&**curent**->**a**.**height**);**

last**->**next **=** curent**;**

last **=** curent**;**

**}**

last**->**next **=** 0**;**

**}**

void output\_struct**(){**

mylist **\***c**;**

c **=** first**;**

int i **=** 1**;**

**while(**c**){**

printf**(**"\n%i furn.:"**,**i**);**

i**++;**

printf**(**"\n length %i"**,**c**->**a**.**length**);**

printf**(**"\n width %i"**,**c**->**a**.**width**);**

printf**(**"\n height %i"**,**c**->**a**.**height**);**

c **=** c**->**next**;**

**}**

printf**(**"\n"**);**

**}**

void delEl**(**int n**){**

mylist **\***c**;**

mylist **\***prev**;**

c **=** first**;**

prev **=** first**;**

int index**=**0**;**

bool b**=false;**

printf**(**"\nEnter the number of furniture, that you want to delete:"**);**

scanf**(**"%i"**,&**index**);**

**if(**index **==** 1**){**

first **=** first**->**next**;**

**}**

**else** **if(**index **==** n**){**

**for(**int i **=** 1**;**i**<**index**-**1**;**i**++){**

c **=** c**->**next**;**

**}**

last **=** c**;**

last**->**next **=** 0**;**

**}**

**else{**

**for(**int i**=**1**;**i**<**index**-**1**;**i**++){**

prev **=** prev**->**next**;**

**}**

c **=** prev**->**next**;**

prev**->**next **=** c**->**next**;**

c**->**next **=** 0**;**

**}**

**}**

void insert\_before**(**int n**)**

**{**

printf**(**"\nType index before to insert a new struct: "**);**

int index**;**

scanf**(**"%i"**,&**index**);**

mylist **\***c**;**

mylist **\***new\_el**;**

**if(**index **==** 1**){**

c **=** **new** mylist**;**

printf**(**" length = "**);**

scanf**(**"%i"**,&**c**->**a**.**length**);**

printf**(**" width = "**);**

scanf**(**"%i"**,&**c**->**a**.**width**);**

printf**(**" height = "**);**

scanf**(**"%i"**,&**c**->**a**.**height**);**

c**->**next **=** first**;**

first **=** c**;**

first **=** c**;**

**}**

//zdes;

**else** **if(**index **==** n**+**1**){**

new\_el **=** first**;**

c **=** **new** mylist**;**

printf**(**" length = "**);**

scanf**(**"%i"**,&**c**->**a**.**length**);**

printf**(**" width = "**);**

scanf**(**"%i"**,&**c**->**a**.**width**);**

printf**(**" height = "**);**

scanf**(**"%i"**,&**c**->**a**.**height**);**

**for(**int i**=**1**;**i**<**index**-**1**;**i**++){**

new\_el **=** new\_el**->**next**;**

**}**

c**->**next **=** new\_el**->**next**;**

new\_el**->**next **=** c**;**

last **=** c**;**

**}**

**else{**

new\_el **=** first**;**

c **=** **new** mylist**;**

printf**(**" length = "**);**

scanf**(**"%i"**,&**c**->**a**.**length**);**

printf**(**" width = "**);**

scanf**(**"%i"**,&**c**->**a**.**width**);**

printf**(**" height = "**);**

scanf**(**"%i"**,&**c**->**a**.**height**);**

**for(**int i**=**1**;**i**<**index**-**1**;**i**++){**

new\_el **=** new\_el**->**next**;**

**}**

c**->**next **=** new\_el**->**next**;**

new\_el**->**next **=** c**;**

**}**

**}**

int main**(){**

SetColor**(**15**);**

int n**=**2**,**e**=**3**,**m**=**0**;**

printf**(**"Type number of furnitures:"**);**scanf**(**"%i"**,&**n**);**

int k**,**menuCase**;**

printf**(**"Enter the size of furnitures:\n"**);**

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

printf**(**"%i furniture:\n"**,**i**+**1**);**

create\_list**(**i**);**

**}**

Menu**:**

printf**(**"\nMenu:\n0)Exit\n1)Output struct\n2)Insert one element to struct\n3)Delete an element in the struct\n-->"**);**

scanf**(**"%i"**,&**menuCase**);**

**switch(**menuCase**){**

**case** 0**:**

**goto** End**;**

**break;**

**case** 1**:**

output\_struct**();**

**goto** Menu**;**

**break;**

**case** 2**:**

insert\_before**(**n**);**

**goto** Menu**;**

**break;**

**case** 3**:**

delEl**(**n**);**

**goto** Menu**;**

**break;**

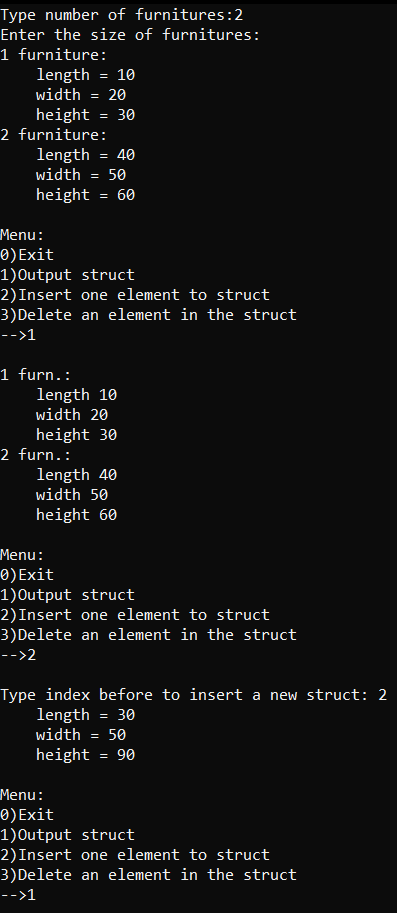
**}**

End**:**

getche**();**

**}**

**Консоль:**

****

****

**Лабораторная работа №20.**

**Задание 1.**

**Условие:**

**Найти решение функции при помощи алгоритма Разделяй и Властвуй.**

**Программа:**

#include <stdio.h>

#include <math.h>

double f**(**double x**){**

**if(**pow**(**x**,**2**)+**2**\***x**-**8**||**pow**(**x**,**2**)+**2**\***x**-**3**){**

**return** 24**/(**pow**(**x**,**2**)+**2**\***x**-**8**)-**15**/(**pow**(**x**,**2**)+**2**\***x**-**3**)-**2**;**

**}**

**return** 1**;**

**}**

double func**(**double a**,**double b**){**

double m**=(**a**+**b**)/**2**;**

**if(**fabs**(**b**-**a**)<=**0.0001**)** **return** m**;**

**if(**f**(**a**)\***f**(**m**)<=**0**)** **return** func**(**a**,**m**);**

**if(**f**(**m**)\***f**(**b**)<=**0**)** **return** func**(**m**,**b**);**

**}**

int main**(){**

printf**(**"\nAnswer:\n"**);**

printf**(**"(-3,-1):%.2f\n"**,**func**(-**3**,-**1**));**

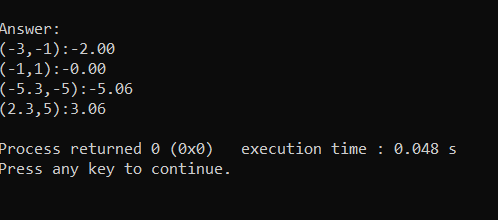
printf**(**"(-1,1):%.2f\n"**,**func**(-**1**,**1**));**

printf**(**"(-5.3,-5):%.2f\n"**,**func**(-**5.3**,-**5**));**

printf**(**"(2.3,5):%.2f\n"**,**func**(**2**,**3.5**));**

**}**

**Консоль:**

****