**Лабораторная работа №8**

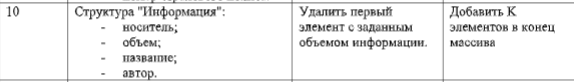
**Разработка программ с использованием структур**

Вариант 10

Цель работы:

Научиться разрабатывать программы с использованием структур с использованием среды Visual Studio.Net.

Задания для самостоятельной работы



namespace ConsoleApp1

{

//Структура Information

struct Information

{

//поля структуры

public string Carrier;

public int Volume;

public string Name;

public string Author;

//метод стуктуры

public void Print()

{

Console.WriteLine($"Carrier: {Carrier, 15}\t" +

$"Volume: {Volume,7}\t" +

$"Name: {Name,15}\t" +

$"Author: {Author,15}\n");

}

}

internal class Program

{

static void Main(string[] args)

{

// массивы с данными

string[] carriers = { "DVD", "CD-ROM", "USB-флешка", "Винчестер", "Магнитная лента", "Оптический диск", "Карта памяти" };

int[] volumes = { 4, 700, 32, 500, 10000, 16, 128 };

string[] names = { "Война и мир", "Гарри Поттер", "Мастер и Маргарита", "Дон Кихот", "Преступление и наказание", "Властелин колец", "Анна Каренина" };

string[] authors = { "Лев Толстой", "Джоан Роулинг", "Михаил Булгаков", "Мигель де Сервантес", "Фёдор Достоевский", "Джон Толкин", "Лев Толстой" };

// массив структур

Information[] information = new Information[carriers.Length];

//заполнение массива данными

for (int i = 0; i < carriers.Length; i++)

{

information[i] = new Information { Carrier = carriers[i], Volume = volumes[i], Name = names[i], Author = authors[i] };

}

//вывод элементов массива на экран

Console.WriteLine("Вывод массива: ");

foreach (Information info in information)

info.Print();

Console.WriteLine("Введите объем: ");

int volume = int.Parse(Console.ReadLine());

DeleteElemByVolume(ref information, volume);//метод, удаляющий элемент из списка по значению поля volume

Console.WriteLine("Массив после удаления элемента\n");

foreach (Information info in information)

info.Print();

//добавление k элементов в массив

Console.WriteLine("Введите K= ");

byte k = byte.Parse(Console.ReadLine());

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

{

Append(ref information, new Information {Carrier=Console.ReadLine(), Volume = int.Parse(Console.ReadLine()), Name= Console.ReadLine(), Author = Console.ReadLine()});

}

Console.WriteLine($"Массив после добавления {k} элементов\n");

foreach (Information info in information)

info.Print();

}

//метод удаляющий элемент из массива

static void DeleteElemByVolume(ref Information[] information, int Volume) {

Information[] timeArray = new Information[information.Length - 1];

int index = 0;

for (int i = 0; i < information.Length; i++) {

if (information[i].Volume == Volume)

{

index = i;

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

timeArray[j] = information[j];

for (int j = index; j < information.Length - 1; j++)

timeArray[j] = information[j + 1];

information = timeArray;

break;

}

}

}

//метод добавляющий элемент в массив

static void Append(ref Information[] array, Information info)

{

Information[] newArr = new Information[array.Length + 1];

for( int i = 0; i < array.Length; i++)

{

newArr[i] = array[i];

}

newArr[newArr.Length - 1] = info;

array = newArr;

}

}

}

