**ЗАДАНИЕ 8**

Задание 1. Описать структуру с именем ZNAK

Написать программу, выполняющую следующие действия:

ввод с клавиатуры данных в массив, состоящий из восьми

элементов типа ZNAK (записи должны быть упорядочены по дате рождения) вывод на экран информации о людях, родившихся под знаком,

название которого введено с клавиатуры (если таких нет, вывести

соответствующее сообщение)..

Листинг программы:

namespace zad1

{

using System;

class Program

{

static void Main(string[] args)

{

ZNAK[] znaks = new ZNAK[2];

// Ввод данных в массив

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

{

Console.WriteLine($"Введите данные для {i + 1}-го человека:");

Console.Write("Фамилия: ");

znaks[i].LastName = Console.ReadLine();

Console.Write("Имя: ");

znaks[i].FirstName = Console.ReadLine();

Console.Write("Знак Зодиака: ");

znaks[i].ZodiacSign = Console.ReadLine();

Console.Write("Дата рождения (день, месяц, год): ");

znaks[i].BirthDate = Array.ConvertAll(Console.ReadLine().Split(','), int.Parse);

}

Console.Write("Введите название знака Зодиака: ");

string zodiacSignToFind = Console.ReadLine();

bool found = false;

foreach (ZNAK znak in znaks)

{

if (znak.ZodiacSign == zodiacSignToFind)

{

Console.WriteLine($"--- {znak.LastName} {znak.FirstName} ({znak.ZodiacSign}) ---");

Console.WriteLine($"Дата рождения: {znak.BirthDate[0]}");

found = true;

}

}

if (!found)

{

Console.WriteLine($"Люди, родившиеся под знаком {zodiacSignToFind}, не найдены.");

}

}

struct ZNAK

{

public string LastName;

public string FirstName;

public string ZodiacSign;

public int[] BirthDate;

}

}

}

Таблица 8.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Данные о людях |  |

Анализ результатов:

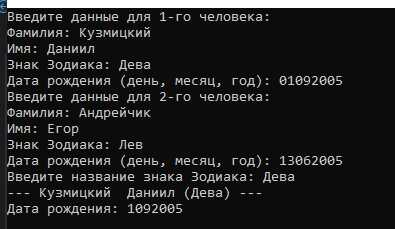


Рисунок 8.1 – Результат работы программы

Задание 2. Создайте перечисление, в котором будут содержаться должности

сотрудников как имена констант. Присвойте каждой константе значение,

задающее количество часов, которые должен отработать сотрудник за месяц.

Листинг программы:

namespace zad2

{

public enum Positions

{

Director = 160,

Manager = 150,

Specialist = 140,

Intern = 120

}

internal class Program

{

static void Main(string[] args)

{

Positions myPosition = Positions.Specialist;

int requiredHours = (int)myPosition;

Console.WriteLine($"Должность: {myPosition}");

Console.WriteLine($"Норма рабочего времени: {requiredHours} часов");

}

}

}

Таблица 8.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Заданы программой | Должность: Specialist  Норма рабочего времени: 140 часов |

Анализ результатов:

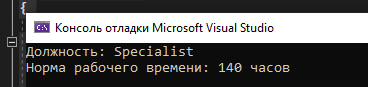


Рисунок 8.2 – Результат работы программы