**8 «ПЕРЕЧИСЛЕНИЯ. СТРУКТУРЫ».**

Задание 1. Описать структуру с именем STUDENT, содержащую следующие поля:

 фамилия и инициалы;

 номер группы;

 успеваемость (массив из пяти элементов).

Написать программу, выполняющую следующие действия:

 ввод с клавиатуры данных в массив, состоящий из десяти структур

типа STUDENT (записи должны быть упорядочены по возрастанию номера

группы);

 вывод на экран фамилий и номеров групп для всех студентов,

включенных в массив, если средний балл студента больше 4,0 (если таких

студентов нет, вывести соответствующее сообщение).

Листинг программы:

using System;

using System.Linq;

struct NOTE

{

public string LastName;

public string FirstName;

public string PhoneNumber;

public int[] BirthDate;

}

class Program

{

static void Main(string[] args)

{

NOTE[] notes = new NOTE[8]

{

new NOTE { LastName = "Иванов", FirstName = "Иван", PhoneNumber = "1234567890", BirthDate = new int[] {1, 1, 1990} },

new NOTE { LastName = "Петров", FirstName = "Петр", PhoneNumber = "2345678901", BirthDate = new int[] {2, 2, 1991} },

new NOTE { LastName = "Сидоров", FirstName = "Сидор", PhoneNumber =

"3456789012", BirthDate = new int[] {3, 3, 1992} },

new NOTE { LastName = "Кузнецов", FirstName = "Кузнец", PhoneNumber = "4567890123", BirthDate = new int[] {4, 4, 1993} },

new NOTE { LastName = "Смирнов", FirstName = "Смирн", PhoneNumber = "5678901234", BirthDate = new int[] {5, 5, 1994} },

new NOTE { LastName = "Васильев", FirstName = "Василий", PhoneNumber = "6789012345", BirthDate = new int[] {6, 6, 1995} },

new NOTE { LastName = "Зайцев", FirstName = "Заяц", PhoneNumber = "7890123456", BirthDate = new int[] {7, 7, 1996} },

new NOTE { LastName = "Медведев", FirstName = "Медведь", PhoneNumber = "8901234567", BirthDate = new int[] {8, 8, 1997} }

};

// Сортировка по трем первым цифрам номера телефона

notes = notes.OrderBy(n => n.PhoneNumber.Substring(0, 3)).ToArray();

Console.WriteLine("Введите фамилию для поиска:");

string searchLastName = Console.ReadLine();

var note = notes.FirstOrDefault(n => n.LastName == searchLastName);

if (note.LastName != null)

{

Console.WriteLine($"Фамилия: {note.LastName}\nИмя: {note.FirstName}\nНомер телефона: {note.PhoneNumber}\nДата рождения: {string.Join(".", note.BirthDate)}");

}

else

{

Console.WriteLine("Человек с такой фамилией не найден.");

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Зайцев | Фамилия: Зайцев  Имя: Заяц  Номер телефона: 7890123456  Дата рождения: 7.7.1996 |

Анализ результатов:

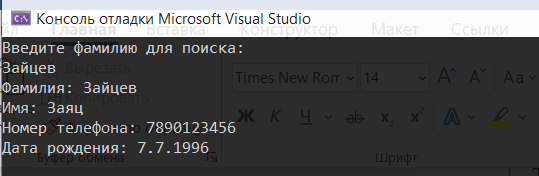


Рисунок 1.1 – Результат работы программы

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:

using System;

using System.Text.RegularExpressions;

class Program

{

static void Main()

{

string input = "Антон работает в Альфа-банке. Его друг Алексей тоже там работает.";

string pattern = @"\b[AaАа]\w\*\b";

MatchCollection matches = Regex.Matches(input, pattern);

Console.WriteLine("Слова, начинающиеся с буквы 'A' или 'а':");

foreach (Match match in matches)

{

Console.WriteLine(match.Value);

}

}

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Директор должен отработать Director часов в месяц.  Менеджер должен отработать Manager часов в месяц.  Инженер должен отработать Engineer часов в месяц.  Клерк должен отработать Clerk часов в месяц.  Уборщик должен отработать Janitor часов в месяц.Алексей |

Анализ результатов:

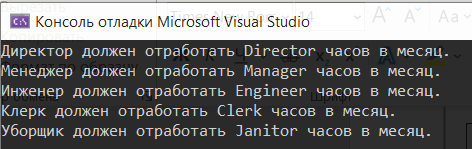


Рисунок 1.2 – Результат работы программы