1. **CТРУКТУРЫ, ПЕРЕЧИСЛЕНИЯ**

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

- фамилия, имя;

- номер телефона;

- дата рождения (массив из трех чисел).

Написать программу, выполняющую следующие действия:

- ввод с клавиатуры данных в массив, состоящий из восьми

элементов типа NOTE (записи должны быть размещены по алфавиту);

- вывод на экран информации о людях, чьи дни рождения

приходятся на месяц, значение которого введено с клавиатуры (если таких нет, вывести соответствующее сообщение).

Листинг программы:

using Note;

try

{

NOTE[] list = new NOTE[2];

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

{

Console.Write("Введите имя: ");

string name = Console.ReadLine();

Console.Write("Введите номер телефона: ");

string phoneNumber = Console.ReadLine();

Console.WriteLine("Введите дату рождения");

int[] birthday = new int[3];

Console.Write("Введите день: ");

birthday[0] = Int32.Parse(Console.ReadLine());

Console.Write("Введите месяц: ");

birthday[1] = Int32.Parse(Console.ReadLine());

Console.Write("Введите год: ");

birthday[2] = Int32.Parse(Console.ReadLine());

list[i] = new(name, phoneNumber, birthday);

}

list = Sort(list);

int choice = 1;

while (choice == 1)

{

Console.Write("Введите месяц рождения: ");

int month = Convert.ToInt32(Console.ReadLine());

int counter = 0;

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

{

if (month == list[i].BirthdayDate[1])

{

list[i].Print();

counter++;

}

}

if (counter == 0)

Console.WriteLine("Нет совпадений");

Console.WriteLine("Хотите продолжить?" +

'\n' + "1. Да" +

'\n' + "2. Нет");

choice = Int32.Parse(Console.ReadLine());

}

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

NOTE[] Sort(NOTE[] list)

{

string tmp = string.Empty;

int[] temp = new int[3];

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

{

for (int j = i + 1; j < list.Length; j++)

{

if (list[i].Name.CompareTo(list[j].Name) > 0)

{

tmp = list[i].Name;

list[i].Name = list[j].Name;

list[j].Name = tmp;

tmp = list[i].PhoneNumber;

list[i].PhoneNumber = list[j].PhoneNumber;

list[j].PhoneNumber = tmp;

temp = list[i].BirthdayDate;

list[i].BirthdayDate = list[j].BirthdayDate;

list[j].BirthdayDate = temp;

}

}

}

return list;

}

Анализ результатов:

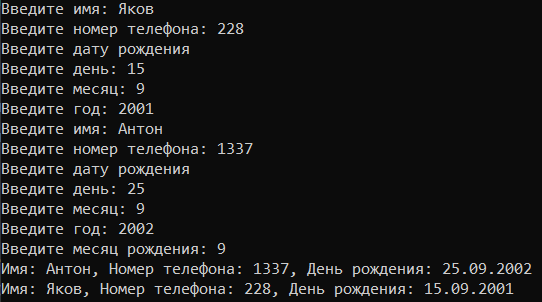


Рисунок 8.1 – Результат работы программы

Источник: собственная разработка

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. Создайте класс Accauntant с методом bool AskForBonus (Post worker, int hours), отражающее давать или нет сотруднику премию. Если сотрудник отработал больше положеных часов в месяц, то ему положена премия.

Листинг программы:

int value = 1;

try

{

while (value == 1)

{

Console.WriteLine("Выберите работника:" +

'\n' + "1. Admin" +

'\n' + "2. Worker" +

'\n' + "3. Manager" +

'\n' + "4. Accountant");

int choice = Int32.Parse(Console.ReadLine());

Accountant acc = new Accountant();

switch (choice)

{

case 1:

acc.worker = Post.Admin;

break;

case 2:

acc.worker = Post.Worker;

break;

case 3:

acc.worker = Post.Manager;

break;

case 4:

acc.worker = Post.Accountant;

break;

}

Console.WriteLine("Введите количество отработанных часов: ");

int hours = Int32.Parse(Console.ReadLine());

if (Accountant.AskForBonus(acc.worker, hours))

Console.WriteLine("Сотруднику положена премия");

else Console.WriteLine("Сотруднику не положена премия");

Console.WriteLine("Хотите продолжить?" +

'\n' + "1. Да" +

'\n' + "2. Нет");

value = Int32.Parse(Console.ReadLine());

}

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

enum Post

{

Admin = 50,

Worker = 40,

Manager = 44,

Accountant = 45

}

class Accountant

{

public Post worker;

public static bool AskForBonus(Post worker, int hours)

{

if ((int)worker <= hours)

return true;

else return false;

}

}

Таблица 8.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 60 часов | Положена премия |

Анализ результатов:

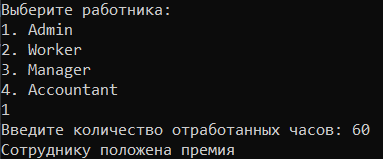


Рисунок 8.2 – Результат работы программы

Источник: собственная разработка