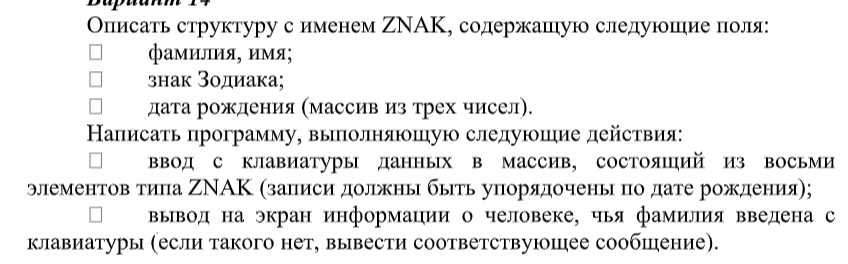
**ПРАКТИЧЕСКОЕ ЗАДАНИЕ 8**

**ПЕРЕЧИСЛЕНИЯ. СТРУКТУРЫ**

Задание 1.



Листинг программы

namespace ConsoleApp1

{

struct ZNAK

{

public string firstname;

public string secondname;

public string zodiac;

public int birthD;

public int birthM;

public int birthY;

public ZNAK(string Firstname, string Secondname, string Zodiac, int BirthD, int BirthM, int BirthY)

{

this.firstname = Firstname;

this.secondname = Secondname;

this.zodiac = Zodiac;

this.birthD = BirthD;

this.birthM = BirthM;

this.birthY = BirthY;

}

public override string ToString()

{

return (string.Format(@"

Имя: {0}

Фама: {1}

Знак: {2}

Дата рождения: {3}.{4}.{5} ", firstname, secondname, zodiac, birthD, birthM, birthY));

}

}

class Program

{

static void Main(string[] args)

{

int a = 8;

ZNAK[] mas = new ZNAK[a];

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

{

Console.Write("Введите имя: ");

mas[i].firstname = Console.ReadLine();

Console.Write("Введите фамилию: ");

mas[i].secondname = Console.ReadLine();

Console.Write("Введите знак зодиака: ");

mas[i].zodiac = Console.ReadLine();

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

mas[i].birthD = int.Parse(Console.ReadLine());

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

mas[i].birthM = int.Parse(Console.ReadLine());

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

mas[i].birthY = int.Parse(Console.ReadLine());

}

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

{

for (int j = 0; j < a - 1 - i; j++)

{

if (mas[i].birthY > mas[j].birthY)

{

ZNAK temp = mas[i];

mas[i] = mas[j];

mas[j] = temp;

}

}

}

Console.WriteLine("Информация о людях (отсортированная по дате рождения): ");

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

{

Console.WriteLine(mas[i]);

}

string flag;

Console.WriteLine("Введите фамилию человека: ");

flag = Console.ReadLine();

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

{

if (mas[i].secondname == flag)

{

Console.WriteLine(mas[i]);

}

else Console.WriteLine("такого челка нет");

break;

}

Console.ReadKey();

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Введите имя:  Введите фамилию:  Введите знак зодиака:  Введите ДЕНЬ рождения:  Введите МЕСЯЦ рождения:  Введите ГОД рождения: | Отсортированный список по дате рождения  Вывод информации о человеке по его фамилии |

Анализ результатов:

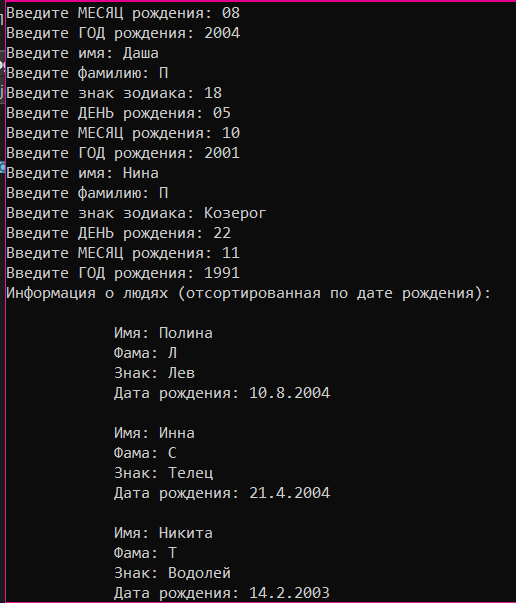


Рисунок 1.1 – Результат работы программы

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:

enum positions

{

manager = 170,

seller = 200,

teacher = 300,

actor = 400,

Baker = 50,

Manager = 401,

Actor = 402,

}

class Program

{

static void Main(string[] args)

{

Console.WriteLine(" All positions");

Console.WriteLine();

Console.WriteLine(" 1. Manager");

Console.WriteLine(" 2. Seller");

Console.WriteLine(" 3. Teacher");

Console.WriteLine(" 4. Actor");

Console.WriteLine(" 5. Baker");

Console.WriteLine();

Console.Write("Choose position: ");

int a = Convert.ToInt16(Console.ReadLine());

positions worker = new positions();

switch (a)

{

case 1:

{

Console.Write("Number of hours worked to receive bonus: ");

Console.WriteLine((int)positions.Manager); worker =positions.Manager;

}

break;

case 2:

{

Console.Write("Number of hours worked to receive bonus: ");

Console.WriteLine((int)positions.seller); worker = positions.seller;

}

break;

case 3:

{

Console.Write("Number of hours worked to receive bonus: ");

Console.WriteLine((int)positions.teacher); worker = positions.teacher;

}

break;

case 4:

{

Console.Write("Number of hours worked to receive bonus: ");

Console.WriteLine((int)positions.actor); worker = positions.Actor;

}

break;

case 5:

{

Console.Write("Number of hours worked to receive bonus: ");

Console.WriteLine((int)positions.Baker); worker = positions.Baker;

}

break;

default: Console.WriteLine("The entered number is not recognized.."); break;

}

Console.WriteLine();

Console.WriteLine("Enter the number of hours worked per month: ");

int h = Convert.ToInt16(Console.ReadLine());

Accountant d = new Accountant();

Console.WriteLine("the presence of a premium ");

Console.WriteLine(d.AskForBonus(worker, h));

Console.ReadLine();

}

}

}

internal class Accountant

{

public bool AskForBonus(positions pos, int hours)

{

if (hours > (int)pos) return true;

else return false;

}

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Choose position: 2  Enter the number of hours worked per month:  250 | the presence of a premium  True |

Анализ результатов:

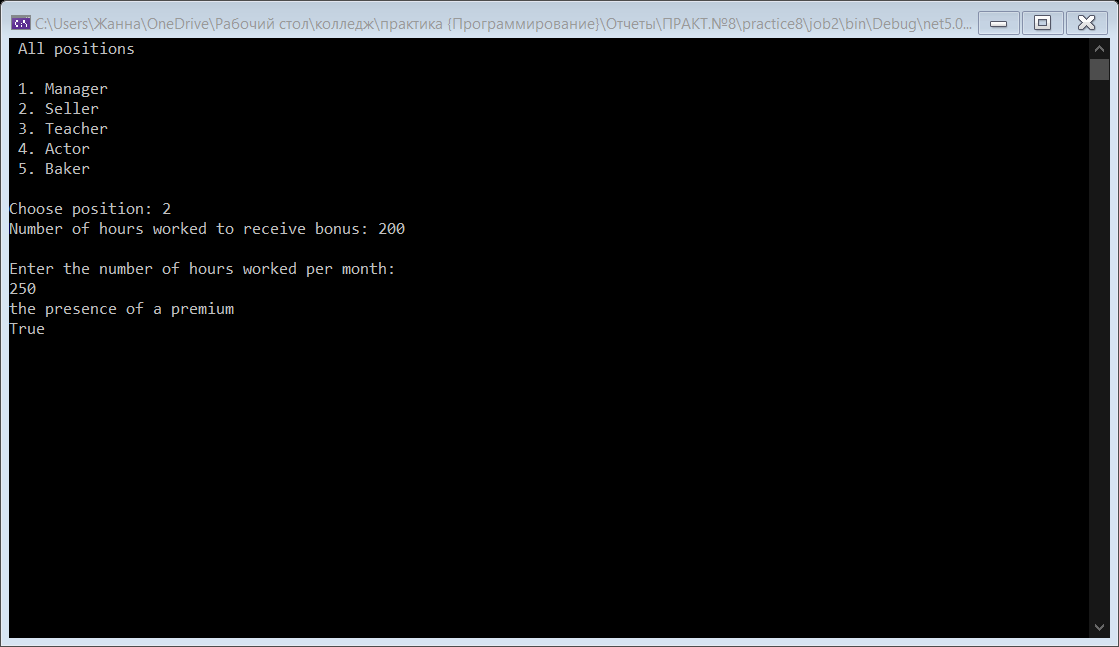


Рисунок 1.2 – Результат работы программы