**8 Структуры, перечисления**

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 среднего балла);

 вывод на экран фамилий и номеров групп для всех студентов, имеющих оценки 4 и 5 (если таких студентов нет, вывести соответствующее сообщение).

Листинг программы:

using System;

using System.Linq;

namespace Numb1

{

struct Student

{

public string familiya; // public модификатор доступа

public int Kolvo;

public double[] ocenki;

public Student(string name, int number, double[] time)

{

this.familiya = name; //this ссылается на текущий экземпляр класса

this.Kolvo = number;

this.ocenki = time;

}

public override string ToString() //Метод override предоставляет новую реализацию метода, унаследованного от базового класса

{

return $"{familiya},{Kolvo},{ocenki}";

}

}

class Program

{

static void Main(string[] args)

{

int count;

Console.WriteLine("Введите количество студентов:");

count = Convert.ToInt32(Console.ReadLine());

Student[] students = new Student[count];

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

{

Console.WriteLine("Введите фамилию и инициалы");

string name = Console.ReadLine();

Console.WriteLine("Введите номер группы");

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

Console.WriteLine("Введите успеваемость (массив из пяти элементов)");

double[] uspevaemost = new double[5];

for (int j = 0; j < 5; j++)

{

uspevaemost[j] = double.Parse(Console.ReadLine());

}

students[i] = new Student(name, number, uspevaemost);

}

Console.WriteLine("Фамилия и инициалы, номер группы, успеваемость (массив из пяти элементов):");

Console.WriteLine("Студенты, имеющие оценки 4 и 5: ");

Boolean bol = true;

foreach (var item in students) //foreach предназначен для перебора элементов массива

{

if (item.ocenki.Min() > 3)

{

Console.WriteLine(item.familiya);

Console.WriteLine(item.Kolvo);

foreach (var otmetki in item.ocenki)

{

Console.Write($"{otmetki} ");

}

bol = false;

}

}

if (bol)

{

Console.WriteLine($"Студенты, имеющие оценки 4 и 5 отсутствуют");

}

Console.ReadLine();}}}

Таблица 8.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Близнюк Е.А.  34  10 10 10 10 10 | Близнюк Е.А.  34  10 10 10 10 10 |

Анализ результатов представлен на рисунке 8.1:

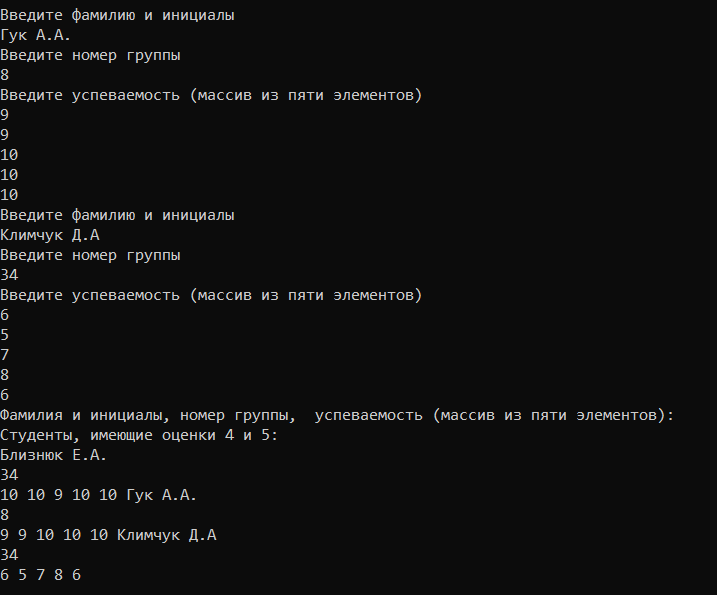


Рисунок 8.1 – Результаты работы программы

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.

Листинг программы:

using static System.Console;

namespace MyNamespace

{

class Program

{

public static void Main()

{

Employees Vladimir = Employees.Vladimir;

Employees vadim = Employees.Vadim;

Employees alex = Employees.Alex;

Employees Jenya = Employees.Jenya;

Employees Arthur = Employees.Arthur;

Employees Serhio = Employees.Serhio;

Employees Maximiliano = Employees.Maximiliano;

string[] employeersNames = { "Vladimir", "vadim", "alex", "Jenya", "Arthur", "serhio", "Maximiliano" };

printAllEmployers(employeersNames);

Console.WriteLine("Введите имя сотрудника о котором хотите узнать информацию(для выходы из программы напишите exit): ");

string employeeChecker = Console.ReadLine();

switch (employeeChecker)

{

case "Vladimir":

EmployeeInformation(Vladimir);

break;

case "vadim":

EmployeeInformation(vadim);

break;

case "alex":

EmployeeInformation(alex);

break;

case "Jenya":

EmployeeInformation(Jenya);

break;

case "Arthur":

EmployeeInformation(Arthur);

break;

case "Serhio":

EmployeeInformation(Serhio);

break;

case "Maximiliano":

EmployeeInformation(Maximiliano);

break;

case "exit:":

break;

}

}

enum Employees

{

Vladimir = 10,

Vadim = 20,

Alex = 5,

Jenya = 15,

Arthur = 19,

Serhio = 30,

Maximiliano = 44,

}

static void EmployeeInformation(Employees employees)

{

switch (employees)

{

case Employees.Vladimir:

Console.WriteLine($"Сотруднику {Employees.Vladimir} требуется отработать:{(int)Employees.Vladimir} часов за месяц");

break;

case Employees.Vadim:

Console.WriteLine($"Сотруднику {Employees.Vadim} требуется отработать:{(int)Employees.Vadim} часов за месяц");

break;

case Employees.Alex:

Console.WriteLine($"Сотруднику {Employees.Alex} требуется отработать:{(int)Employees.Alex} часов за месяц");

break;

case Employees.Jenya:

Console.WriteLine($"Сотруднику {Employees.Jenya} требуется отработать:{(int)Employees.Jenya} часов за месяц");

break;

case Employees.Arthur:

Console.WriteLine($"Сотруднику {Employees.Arthur} требуется отработать:{(int)Employees.Arthur} часов за месяц");

break;

case Employees.Serhio:

Console.WriteLine($"Сотруднику {Employees.Serhio} требуется отработать:{(int)Employees.Serhio} часов за месяц");

break;

case Employees.Maximiliano:

Console.WriteLine($"Сотруднику {Employees.Maximiliano} требуется отработать:{(int)Employees.Maximiliano} часов за месяц");

break;

}

}

static void printAllEmployers(string[] arr)

{

Console.WriteLine("Все сотрудники:");

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

{

Console.WriteLine(i + "." + arr[i]);

}

}

}

}

Таблица 8.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| vadim | Сотруднику Vadim требуется отработать 20 часов за месяц. |

Анализ результатов представлен на рисунке 8.2:

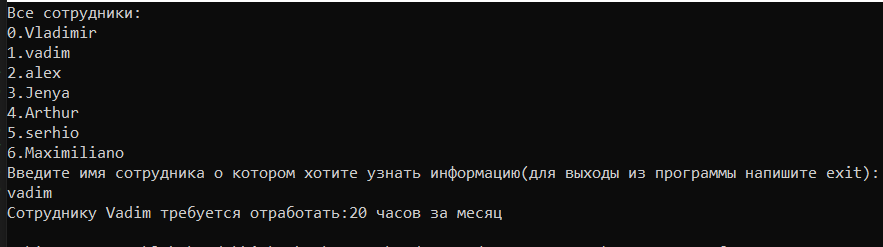


Рисунок 8.2 – Результаты работы программы

Источник: собственная разработка