

8 Перечисления. Структуры

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

название пункта назначения;

номер поезда;

время отправления.

Написать программу, выполняющую следующие действия:

ввод с клавиатуры данных в массив, состоящий из восьми элементов типа TRAIN (записи должны быть размещены в алфавитном порядке по названиям пунктов назначения);

вывод на экран информации о поездах, направляющихся в пункт, название которого введено с клавиатуры (если таких поездов нет, вывести соответствующее сообщение).

Листинг программы:

```
namespace MySpace
{
    public struct Train : IComparable
    {
        public string destination;
        public int number;
        public DateTime departure;

        public Train(string destination, int number, DateTime departure)//Создание
структуры
        {
            this.destination = destination;
            this.number = number;
            this.departure = departure;
        }

        public int CompareTo(object obj)//Сортировка
        {
            Train a = (Train)obj;
            if (this.departure.CompareTo(a.departure) == -1) return -1;
            if (this.departure.CompareTo(a.departure) == 0) return 0;
            else return 1;
        }
    }
}
```

					УП 2-40 01 01.37ТП.227.23.08			
Изм.	Лист	№ док	Подпись	Дата				
Разраб.		Сорокина Е.А.			Перечисления. Структуры		Лит	Лист
Проверил.		Новик А.И.						32
							Гродненский ГКТТид	
Н.контр.								
Утвердил.								

```

    }

    public override string ToString()//Вывод поездов
    {
        return string.Format("Пункт назначения: {0} \nНомер поезда: {1}
\nОтправление: {2:t}", destination, number, this.departure);
    }
}
class Program
{
    static void Main(string[] args)
    {
        Train[] trains = new Train[2];

        for (int i = 0; i < trains.Length; i++)//Ввод поездов
        {
            Console.WriteLine("Введите название пункта назначения: ");
            string destination = Console.ReadLine();

            Console.WriteLine("Введите номер поезда: ");
            int number = int.Parse(Console.ReadLine());

            Console.WriteLine("Введите время отправления: ");
            DateTime departure = DateTime.Parse(Console.ReadLine());

            trains[i] = new Train(destination, number, departure);
        }

        Console.WriteLine("\nВведите название пункта назначения: ");//Поиск
пункта
        string input;
        while ((input = Console.ReadLine()) != " ")
        {
            bool trainFound = false;
            for (int i = 0; i < trains.Length; i++)
            {
                if (trains[i].destination == input)
                {
                    Console.WriteLine(trains[i]);
                    trainFound = true;//Пункт найден
                }
            }

            if (!trainFound)

```

```

        Console.WriteLine("Ничего не найдено. Повторите
попытку");//Пункт не найден
    }
}
}
}

```

Таблица 8.1 – Входные и выходные данные

Входные данные	Выходные данные
Барановичи, 5, 13:00 Минск, 25, 12:50 Минск	Минск, 25, 12:50

Анализ результатов:

```

E:\Учёба\Практика по пр\Les9\Task1\bin\Debug\net6.0\Task1.exe
Введите название пункта назначения:
Барановичи
Введите номер поезда:
5
Введите время отправления:
13:00
Введите название пункта назначения:
Минск
Введите номер поезда:
25
Введите время отправления:
12:50

Введите название пункта назначения:
Минск
Пункт назначения: Минск
Номер поезда: 25
Отправление: 12:50

```

Рисунок 8.1 – Результат работы программы

Задание 2. 1. Создайте перечисление, в котором будут содержаться должности сотрудников как имена констант. Присвойте каждой константе значение, задающее количество часов, которые должен отработать сотрудник за месяц.

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

Листинг программы:

```

namespace MySpace
{
    static class Accountant

```

```

{
    public enum CompanyEmployees
    {
        ДиректорПочты = 100,
        Юрист = 150,
        Экономист = 200,
        Почтальон = 220,
        ОператорПочтовойСвязи = 150
    }

    public static bool AskForBonus(CompanyEmployees postWorker, int hours)
    {
        return hours > (int)postWorker;
    }
}

class Program
{
    static void Main(string[] args)
    {
        CheckBonus(Accountant.CompanyEmployees.ДиректорПочты, 67);
        CheckBonus(Accountant.CompanyEmployees.Юрист, 220);
        CheckBonus(Accountant.CompanyEmployees.Экономист, 300);
        CheckBonus(Accountant.CompanyEmployees.Почтальон, 250);
        CheckBonus(Accountant.CompanyEmployees.ОператорПочтовойСвязи, 280);
        Console.ReadLine();
    }

    static void CheckBonus(Accountant.CompanyEmployees postWorker, int
hours)
    {
        string answer = string.Format("Должность работника: {0} \nЧасы
работы за месяц: {1} \nВыдача бонуса: ", postWorker, hours);
        answer += Accountant.AskForBonus(postWorker, hours) ? "Выдать
премию" : "Не выдавать премию";
        Console.WriteLine(answer);
    }
}

```

Анализ результатов:

```
cs E:\Учёба\Практика по пр\Les9\Task2\bin\Debug\net6.0\Task2.exe
Должность работника: ДиректорПочты
Часы работы за месяц: 67
Выдача бонуса: Не выдавать премию
Должность работника: ОператорПочтовойСвязи
Часы работы за месяц: 220
Выдача бонуса: Выдать премию
Должность работника: Экономист
Часы работы за месяц: 300
Выдача бонуса: Выдать премию
Должность работника: Почтальон
Часы работы за месяц: 250
Выдача бонуса: Выдать премию
Должность работника: ОператорПочтовойСвязи
Часы работы за месяц: 280
Выдача бонуса: Выдать премию
```

Рисунок 8.2 – Результат работы программы