Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №4

за 1 семестр

По дисциплине: «СПП»

Выполнил:

Студент 3 курса

Группы ПО-5

Лозейко П. А.

Проверил:

Крощенко А. А.

2021

Вариант 9



**Код программы:**

**Program.cs**

using System;

namespace lab4.\_1.\_9

{

class Program

{

static void Main(string[] args)

{

Mobile.IPhone phone1 = new Mobile.IPhone();

phone1.Model = "IPhone 13";

phone1.Properties = "SoC Apple A15 Bionic (6 процессорных ядер: 2 высокопроизводительных и 4 энергоэффективных, 4 графических ядра, 16 ядер Neural Engine). Сенсорный дисплей 6,1″, OLED, 2532×1170, 460 ppi, емкостной, мультитач / 5,4″, OLED, 2340×1080, 476 ppi, емкостной, мультитач. RAM(по информации Geekbench 5): 3,6 ГБ. Флэш - память 128 / 256 / 512 ГБ. Поддержка карт памяти отсутствует.Камеры: фронтальная(12 Мп, видео 4К 30 к / с, 720р 240 к / с) и тыльные модули 12 Мп(съемка видео 4К 60 к / с): широкоугольный и сверхширокоугольный";

Mobile.IPhone phone2 = new Mobile.IPhone();

phone1.Print();

phone2.FindProperties("2 ядра");

}

}

}

**Mobile.cs**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4.\_1.\_9

{

class Mobile

{

public class IPhone

{

public string Model { get; set; }

public string Properties { get; set; }

public IPhone()

{

Model = "IPhone 6";

Properties = "SoC Apple A8 @1,4 ГГц (2 ядра, 64-битная архитектура ARMv8-A). Сопроцессор движения Apple M8 (включает акселерометр, гироскоп и компас). GPU PowerVR GX6650 (предположительно). RAM 1 ГБ. Флэш - память 16 / 64 / 128 ГБ. Поддержка карт памяти microSD отсутствует. Операционная система iOS 8.0. Сенсорный дисплей IPS, 4,7″, 1334×750(326 ppi), емкостной, мультитач. Камеры 8 Мп с размером пикселя 1,5 мкм и диафрагмой ƒ/ 2,2(съемка видео Full HD 30 или 60 к / с) и 1,2 Мп с диафрагмой ƒ/ 2,2(съемка видео — 720р)";

}

public void Print()

{

Console.WriteLine($"\tМодель: {Model}\n\tСвойства: {Properties}\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

public void FindProperties(string mainWord)

{

if(Properties.Contains(mainWord))

{

Print();

}

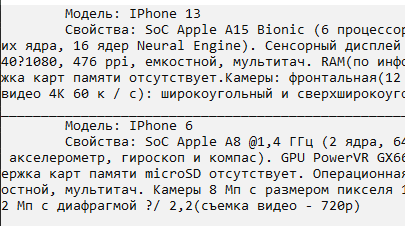
}

}

}

}

**Результат работы программы:**

****

****

**Код программы:**

**Program.cs**

using System;

namespace lab4.\_2.\_9

{

class Program

{

static void Main(string[] args)

{

Wheel Continental = new Wheel(15, 220);

Car Porsche = new Car(Continental);

Porsche.Print();

}

}

}

**Car.cs**

using System;

using System.Collections.Generic;

namespace lab4.\_2.\_9

{

class Wheel

{

public int radius;

public int width;

public Wheel(int r, int w)

{

radius = r;

width = w;

}

}

class Car

{

string model = "Porsche Panamera";

Wheel wheel;

public Car(Wheel someWheel)

{

wheel = someWheel;

}

public void Print()

{

Console.WriteLine($"Модель: {model}. Колёса: радиус - {wheel.radius}, ширина - {wheel.width}");

}

public void ChangeWheel(int r, int w)

{

wheel.radius = r;

wheel.width = w;

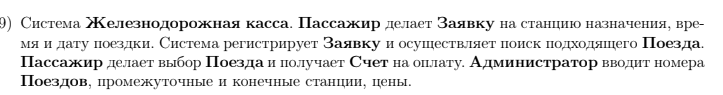
}

}

}

**Результат работы программы:**

****

****

**Program.cs**

using System;

using System.Collections.Generic;

namespace lab4.\_3.\_9

{

class Program

{

static void Main(string[] args)

{

List<Train> Trains = new List<Train>();

Train train1 = new Train(1);

Trains.Add(train1);

Train train2 = new Train(2);

Trains.Add(train2);

Train train3 = new Train(3);

Trains.Add(train3);

Passenger Tom = new Passenger();

DateTime tom = new DateTime(2021, 10, 10);

Tom.Application("Брест", "Минск", tom, Trains);

Tom.TrainSelection(1, Trains);

Administrator Bob = new Administrator();

string[] stations = { "Гродно", "Щучин", "Лида", "Воложин", "Минск" };

DateTime[] dateTimes2 = new DateTime[]

{

new DateTime(2021, 10, 10, 8, 10, 0),

new DateTime(2010, 10, 1, 8, 32, 1),

new DateTime(2010, 10, 1, 8, 45, 2),

new DateTime(2010, 10, 1, 9, 0, 3),

new DateTime(2010, 10, 1, 9, 20, 4),

};

Bob.AddTrain(Trains, 4, stations, dateTimes2, 20);

Trains[3].Print();

}

}

}

**Train.cs**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

namespace lab4.\_3.\_9

{

class Train

{

public List<string> Stations = new List<string>();

public List<DateTime> TrainTime = new List<DateTime>();

public int Price { get; set; }

public int ID { get; set; }

public Train()

{

}

public Train(int Number)

{

string Line = File.ReadLines("trains.txt").ElementAtOrDefault(--Number);

if(Line == null)

{

return;

}

string[] info = Line.Split(" ", StringSplitOptions.RemoveEmptyEntries);

ID = int.Parse(info[0]);

Price = 10;

DateTime time;

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

{

if (i % 2 == 0)

{

time = DateTime.ParseExact(info[i], "dd.MM.yyyy\_HH:mm", null);

TrainTime.Add(time);

}

else

{

Stations.Add(info[i]);

}

}

}

public void Print()

{

Console.WriteLine($"Номер поезда - {ID}, станция отправления - {Stations[0]}, время отправления - {TrainTime[0]}, конечная станция - {Stations[^1]}, время прибытия - {TrainTime[^1]}");

}

}

}

**Passenger.cs**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4.\_3.\_9

{

class Passenger

{

int Bill { get; set; }

public Passenger()

{

Bill = 100;

}

public void Application(string DepartureStation, string DestinationStation, DateTime date, List<Train> train)

{

for(int i = 0; i < train.Count; i++)

{

int index1 = train[i].Stations.IndexOf(DepartureStation);

int index2 = train[i].Stations.IndexOf(DestinationStation);

if (index1 != -1 && index2 != -1)

{

if (index1 < index2)

{

if (train[i].TrainTime[index1] > date)

{

train[i].Print();

continue;

}

}

}

}

}

public void TrainSelection(int Num, List<Train> train)

{

Num--;

if(Bill - train[Num].Price < 0)

{

Console.WriteLine("На счёте на хватает средств!");

return;

}

Bill -= train[Num].Price;

Console.WriteLine($"Остаток на счёте {Bill}");

}

}

}

**Administrator.cs**

using System;

using System.Collections.Generic;

namespace lab4.\_3.\_9

{

class Administrator

{

public void AddTrain(List<Train> trains, int id, string[] Stations, DateTime[] dateTimes, int price)

{

Train train = new Train();

train.ID = id;

train.Price = price;

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

{

train.Stations.Add(Stations[i]);

train.TrainTime.Add(dateTimes[i]);

}

trains.Add(train);

}

}

}

**Результат работы программы:**

****