**11 Полиморфизм**

**Задание №2:** Составить программу с абстрактным родительским классом и двумя объектами - потомками. Организовать вычисление суммарного расхода горючего.

Листинг программы:

using System;

using System.Collections.Generic;

abstract class Car

{

public string Name { get; set; }

public Car(string name) => Name = name;

public abstract double CalculateFuelConsumption();

public virtual void PrintInfo() => Console.WriteLine($"Автомобиль: {Name}\nТип: {GetType().Name}\n{(this is Truck t ? $"Грузоподъемность: {t.LoadCapacity} т" : $"Объем двигателя: {((PassengerCar)this).EngineVolume} см^3")}\nРасход топлива: {CalculateFuelConsumption():F2} л/100км\n----------------------");

}

class Truck : Car

{

public double LoadCapacity { get; set; }

public Truck(string name, double loadCapacity) : base(name) => LoadCapacity = loadCapacity;

public override double CalculateFuelConsumption() => Math.Sqrt(LoadCapacity) \* 100;

}

class PassengerCar : Car

{

public double EngineVolume { get; set; }

public PassengerCar(string name, double engineVolume) : base(name) => EngineVolume = engineVolume;

public override double CalculateFuelConsumption() => 2.5 \* EngineVolume;

}

class Program

{

static void Main()

{

var cars = new List<Car>

{

new Truck("Грузовик-1", 10),

new PassengerCar("Седан-1", 2.0),

new Truck("Грузовик-2", 15),

new PassengerCar("Хэтчбек-1", 1.6),

new Truck("Грузовик-3", 20),

new PassengerCar("Внедорожник-1", 3.5)

};

Console.WriteLine("=== ПРОТОКОЛ РАСЧЕТА РАСХОДА ТОПЛИВА ===\nВсего автомобилей: " + cars.Count + "\n");

cars.ForEach(c => c.PrintInfo());

double total = 0;

cars.ForEach(c => total += c.CalculateFuelConsumption());

Console.WriteLine($"Средний расход всех автомобилей: {total / cars.Count:F2} л/100км");

}

}

Анализ результатов:

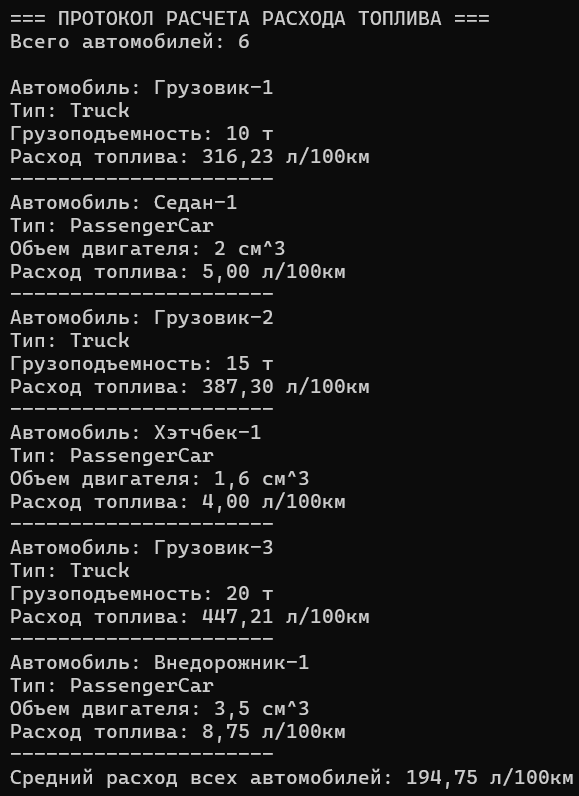


Рисунок 11.1 – Результат работы программы