Министерство образования и науки Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение науки и высшего образования**

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

**Лабораторная работа № 1**

**по дисциплине**

**«Программирование компьютерной графики»**

**Тема: «C#. Разработка консольного приложения»**

Выполнил:

ст. гр. ПРИ-117

Хлызова В.Г.

Принял:

Жигалов И.Е.

Владимир, 2020

ЦЕЛЬ РАБОТЫ

Освоение принципов программирования в среде Microsoft Visual C# для операционной системы Windows с использованием Microsoft .net Framework. Получение практических навыков построения консольных приложений в Visual C#.

ЗАДАНИЕ

1. Ознакомиться по методическим указаниям и литературе с назначением среды VisualC#, ее компонентами и особенностями работы в ней.
2. Выполнить действия, приведенные в разделах 1.2, 1.3 для построения простой консольной программы, изучить ее структуру. При разработке программы имя проекта, создаваемого в MS Visual Studio, должно содержать фамилию студента и группу (например, Ivanov\_Ivan\_ISG\_105\_lab\_1).
3. Дополнить программу, добавив в реализуемый класс (раздел 1.3) два новых метода / команды, управляющих поведением объекта.

На экран должны выводиться текстовые строки, поясняющие работу программы в данном режиме и содержащие ФИО и группу автора.

ХОД РАБОТЫ

Текс разработанной программы:

using System;

namespace Khlyzova\_Valeria\_PRI\_117\_lab\_01

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Лабораторную работу выполнил");

Console.WriteLine("студент группы ПРИ-117");

Console.WriteLine("Хлызова Валерия");

string user\_command = "";

bool Infinity = true;

Man hero = null;

while (Infinity)

{

Console.WriteLine("\nПожалуйста, введите команду");

user\_command = Console.ReadLine();

switch (user\_command)

{

case "help":

{

Console.WriteLine("\nСписок команд:");

Console.WriteLine("---");

Console.WriteLine("Основные команды: ");

Console.WriteLine("create\_man : команда создает человека, (экземпляр класса Man)");

Console.WriteLine("info : команда выводит на экран информацию о человеке (если создан экземпляр класса)");

Console.WriteLine("kill\_man : команда убивает человека (если создан экземпляр класса)");

Console.WriteLine("talk : команда застравляет человека говорить (если создан экземпляр класса)");

Console.WriteLine("go : команда застравляет человека идти (если создан экземпляр класса)\n");

Console.WriteLine("Команды при сражении: ");

Console.WriteLine("hit : нанести удар противнику");

Console.WriteLine("run : попытаться убежать от противника");

Console.WriteLine("yes : пощадить противника");

Console.WriteLine("no : уничтожить противника(((\n");

Console.WriteLine("---");

break;

}

case "create\_man":

{

Console.WriteLine("Пожалуйста, введите имя создаваемого человека");

user\_command = System.Console.ReadLine();

hero = new Man(user\_command);

Console.WriteLine("Человек успешно создан, его имя " + hero.getName() + "\n");

break;

}

case "info":

{

if (hero != null) hero.Info();

else Console.WriteLine("Человек не создан.");

break;

}

case "kill\_man":

{

if (hero != null) hero.Kill();

else Console.WriteLine("Человек не создан. Вы не можете его убить");

break;

}

case "talk":

{

if (hero != null && hero.IsAlive()) Console.WriteLine("Человек не создан. Команда не может быть выполнена");

else Console.WriteLine("Человек не создан. Команда не может быть выполнена");

break;

}

case "go":

{

if (hero != null) hero.Go();

else Console.WriteLine("Человек не создан. Команда не может быть выполнена");

break;

}

case "exit":

{

Infinity = false;

Console.WriteLine("Приграмма завершена");

break;

}

default:

{

Console.WriteLine("Ваша команда не определена, пожалуйста повторите снова");

Console.WriteLine("Для вывода списка команд введите команду help");

Console.WriteLine("Для завершения программы введите команду exit");

break;

}

}

}

}

}

}

Текс класса Man:

using System;

using System.Dynamic;

namespace Khlyzova\_Valeria\_PRI\_117\_lab\_01

{

class Man

{

private Random rnd = new Random();

public Man(string \_name)

{

Name = \_name;

isLife = true;

Age = (uint)rnd.Next(15, 50);

Health = (uint)rnd.Next(10, 100);

}

private string Name;

private uint Age;

private uint Health;

private bool isLife;

public string getName()

{

return Name;

}

public void Talk()

{

int random\_talk = rnd.Next(1, 3);

string tmp\_str = "";

switch (random\_talk)

{

case 1:

{

tmp\_str = "Привет, меня зовут " + Name + ", рад познакомиться";

break;

}

case 2:

{

tmp\_str = "Мне " + Age + ". А тебе?";

break;

}

case 3:

{

if (Health > 50)

tmp\_str = "Да я зводоров как бык!";

else

tmp\_str = "Со здоровьем у меня хреново, дожить бы до " + (Age + 10).ToString();

break;

}

}

System.Console.WriteLine(tmp\_str);

}

public void Go()

{

if (IsAlive())

{

if (Health > 40)

{

string outString = Name + " мирно прогуливается по городу";

Console.WriteLine(outString);

}

else

{

string outString = Name + " болен и не может гулять по городу";

Console.WriteLine(outString);

}

// Случайным образом генерируется схватка с противником

if (rnd.Next() % 2 == 0) Fight();

}

else

{

string outString = Name + " не может идти, он умер";

Console.WriteLine(outString);

}

}

public bool IsAlive()

{

return isLife;

}

public void Kill()

{

isLife = false;

Health = 0;

Console.WriteLine(Name + " умер");

}

// Метод выводит информацию о созданном человеке

public void Info()

{

Console.WriteLine("Имя: " + Name);

Console.WriteLine("Возраст: " + Age);

Console.WriteLine("Здоровье: " + Health);

}

// Бой с противником

private void Fight()

{

// Создаем противника со случайным значением здоровья

uint enemy = (uint)rnd.Next(20, 70);

// Выводим сообщения об атаке и текущее здоровье человека и противника

Console.WriteLine("На человека нападает противник");

Console.WriteLine("Здоровье человека: " + Health + ", здоровье противника: " + enemy);

// Создаем переменную для определения вводимых пользователем команд

string user\_command = "";

// Создаем цекл для схватки

while ((Health >= 10) && (enemy > 10) && (enemy != 100))

{

// Каждую итерацию записываем команду и в зависимости от выбранного действия формируем ход сражения

Console.WriteLine("\nВведите коматду hit чтобы нанести удар, или run чтобы убежать");

user\_command = Console.ReadLine();

switch (user\_command)

{

case "hit":

{

// При аттаки человек и противник наносят поочередно удары, после выводится текущее состояние каждого

Health -= (uint)10;

enemy -= (uint)10;

Console.WriteLine("Человек наносит противнику удар, противник отвечает");

Console.WriteLine("Здоровье человека: " + Health + ", здоровье противника: " + enemy);

break;

}

case "run":

{

// При попытке человека убежат противник решает напасть только если здоровье человека превышает 20 едениц

if (Health < 20)

{

Console.WriteLine("Противник проявил милосердие");

enemy = 100;

}

else

{

Health -= 10;

Console.WriteLine("Противник наносит удар");

Console.WriteLine("Здоровье человека: " + Health + ", здоровье противника: " + enemy);

}

break;

}

default:

{

// Обработка неверно введенной команды

Console.WriteLine("Ваша команда не определена, пожалуйста повторите снова");

Console.WriteLine("Введите коматду hit чтобы нанести удар, или run чтобы убежать");

break;

}

}

}

if (Health <= 10)

{

// Если здоровье человека меньше 10 и он не пытался убежать, противник наносит решающий удар

Console.WriteLine("Противник наносит решающий удар");

Kill();

}

else

{

// Если противник проиграл в честной схватке, он попытается убежать

if (enemy < 100) {

Console.WriteLine("Противник убегает. Проявить милосердие? (Введите команду yes или no)");

user\_command = Console.ReadLine();

switch (user\_command)

{

case "yes":

{

// Если человек пощадил противника его здоровье увеличивается

Health = (uint)rnd.Next((int)Health, 100);

break;

}

case "no":

{

// Если человек не пощадил противника его здоровье увеличиваетсяв, но в меньшем размере

Health = (uint)rnd.Next((int)Health, 80);

break;

}

default:

{

// Обработка неверно введенной команды

Console.WriteLine("Ваша команда не определена, пожалуйста повторите снова");

Console.WriteLine("Введите команду yes или no");

break;

}

}

}

// После сражения выводится информация о текущем здоровье человека

Console.WriteLine("После встречи с противником ваше здоровье составляет " + Health);

}

}

}

}

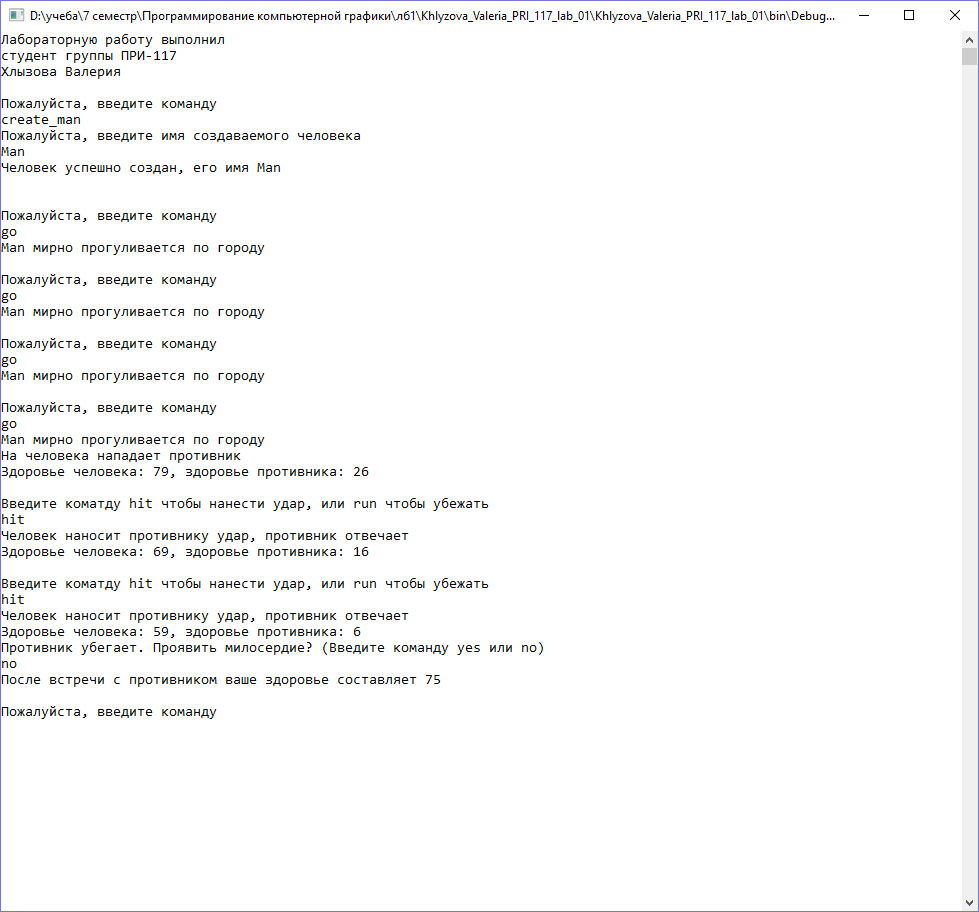
Результат выполнения: 

Рисунок 1. Первый сценарий.

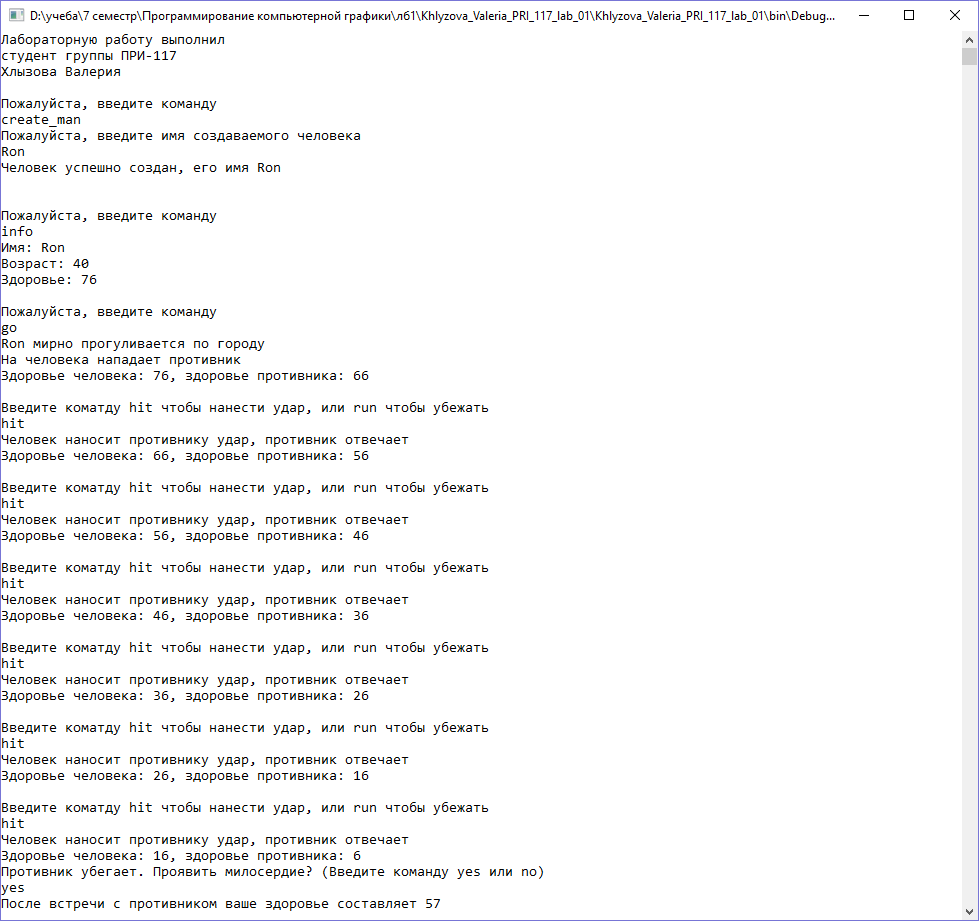


Рисунок 3. Вывод информации о созданном человеке и второй сценарий битвы.

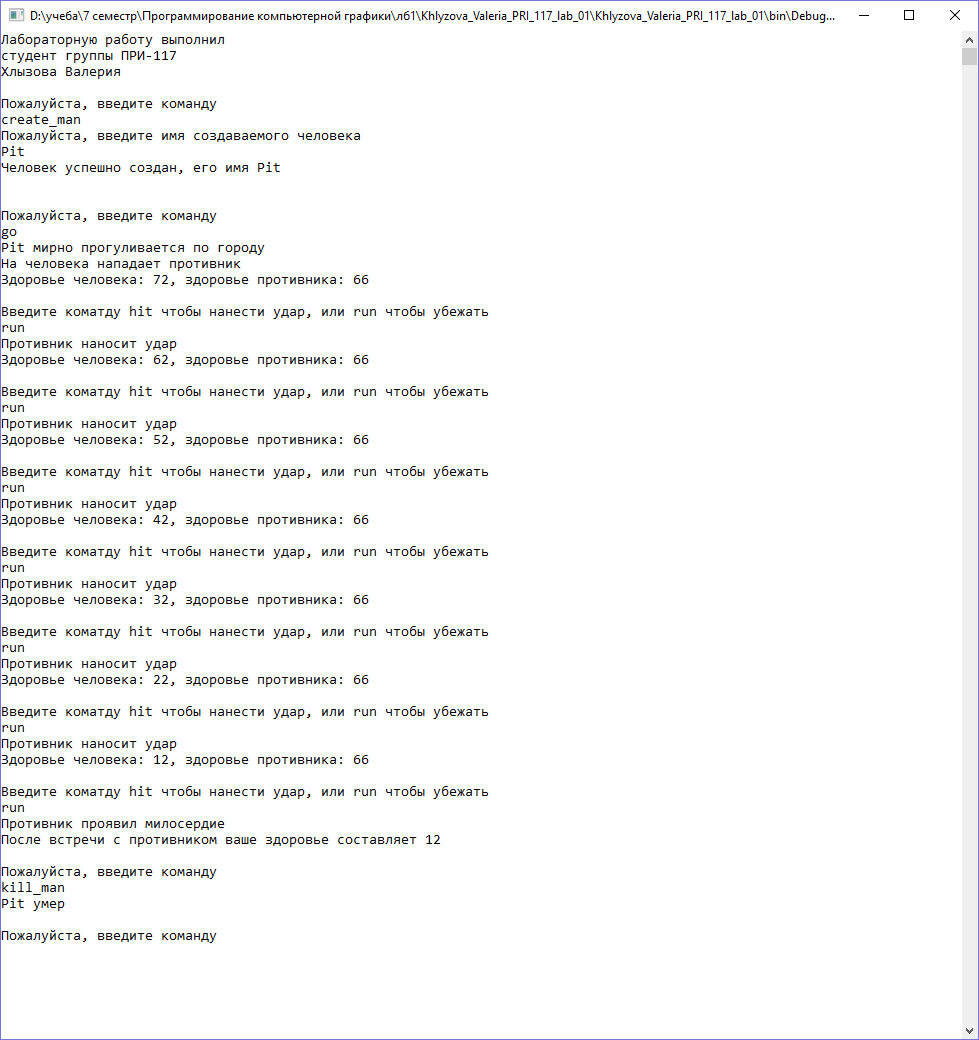


Рисунок 4. Третий сценарий.

ВЫВОД

В ходе выполнения лабораторной работы были освоены принципы программирования в среде Microsoft Visual C# для операционной системы Windows с использованием Microsoft .net Framework. Получены практические навыки построения консольных приложений