

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВВГУ»)
ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

ТЕХНИЧЕСКОЕ ЗАДАНИЕ ТЕКСТОВАЯ RPG-ИГРА

По дисциплине
«Информатика и программирование»

Студент
гр. БИН-25-2 _____ Д.В. Тимкин
Ассистент
преподавателя _____ М.В. Водяницкий

Техническое задание - Текстовая RPG-игра

Вы работаете программистом в небольшой японской компании на заре игровой индустрии. Компания разрабатывает свою первую экспериментальную игру - текстовую RPG, которая должна запускаться прямо в консоли и погружать игрока в атмосферу подземелий, опасностей и развития персонажа

Ваша задача - реализовать прототип игры, который демонстрирует основные игровые механики: характеристики персонажа, бои, прокачку, инвентарь и случайные события

1. Общая идея программы

Программа представляет собой консольную текстовую RPG, в которой игрок:

- 1) создает персонажа (выбор расы)
- 2) получает случайные характеристики в рамках выбранной расы
- 3) исследует подземелье, состоящее из случайных комнат
- 4) сражается с врагами, находит предметы и улучшает персонажа
- 5) повышает уровень и распределяет очки характеристик
- 6) принимает решения, влияющие на дальнейший путь

Игра работает в пошаговом режиме и управляется вводом команд с клавиатуры

2. Создание персонажа

2.1 Выбор расы

В начале игры пользователь выбирает расу персонажа (например):

- 1) Человек
- 2) Эльф
- 3) Дворф

Каждая раса задает диапазоны генерации характеристик

2.2 Характеристики персонажа

Характеристики генерируются случайным образом при создании персонажа, но в допустимых пределах для выбранной расы

Пример набора характеристик (можно расширять):

- 1) **HP** - здоровье
- 2) **Attack** - сила атаки
- 3) **Defense** - защита
- 4) **Agility** - ловкость (влияет на уклонение)

5) **Height** - рост

6) **Weight** - вес

Допускается, что некоторые характеристики влияют друг на друга (например, рост и вес влияют на уклонение или скорость)

3. Опыт и уровни

1) Персонаж получает опыт за победу над врагами

2) При накоплении нужного количества опыта повышается уровень

3) Каждый новый уровень дает очки прокачки

3.1 Прокачка характеристик

Игрок может распределять очки вручную между характеристиками

Пример:

1) +1 к атаке

2) +2 к HP

3) +1 к ловкости

Распределение очков выполняется в комнатах отдыха

4. Инвентарь и экипировка

4.1 Инвентарь

Инвентарь хранит предметы:

1) зелья (лечение и др.)

2) монеты

3) оружие 4) прочие предметы

Игрок может:

1) просматривать инвентарь

2) использовать предметы

3) выбрасывать любые предметы

4.2 Экипировка

В инвентаре должны быть отдельные слоты:

1) оружие

2) броня

Экипированные предметы влияют на характеристики персонажа

5. Подземелье и комнаты

5.1 Структура подземелья

- 1) Игра начинается в подземелье
- 2) Подземелье состоит из комнат
- 3) После каждой комнаты игрок выбирает путь:
- 4) налево
- 5) направо

Развилка есть после каждой комнаты

5.2 Типы комнат

Комнаты генерируются случайно:

- 1) Боевая комната - бой с врагом
- 2) Комната отдыха - без событий
- 3) Комната с сундуком - предметы или золото

Возможны комбинации:

- 1) слева враг, справа сундук
- 2) оба врага
- 3) обе комнаты отдыха

5.3 Видимость комнат

Перед выбором направления игрок:

- 1) иногда знает, что находится дальше
- 2) иногда не знает (темно, неизвестно)

Информация о видимости определяется случайно.

6. Враги и сложность

- 1) Враги генерируются случайно
- 2) У врагов есть характеристики (HP, атака, защита и т.д.)
- 3) С каждым этажом подземелья сложность возрастает
- 4) Каждые 'N' комнат или действий происходит переход на новый этаж

7. Боевая система

Бой происходит в пошаговом режиме:

Пример действий игрока:

- 1) атаковать

2) использовать предмет 3) попытаться уклониться

Учитываются:

- 1) характеристики игрока
- 2) экипировка
- 3) случайные факторы (уклонение, критический удар)

8. Предметы и добыча

- 1) Враги и сундуки могут давать:
- 2) зелья
- 3) оружие
- 4) другие предметы
- 5) Полученные предметы добавляются в инвентарь
- 6) При нехватке места игрок решает, что выбросить

9. Хранение данных

Допускается (но не обязательно):

- 1) сохранение состояния игры в файл
- 2) использование формата **JSON** для хранения:
- 3) характеристик персонажа
- 4) инвентаря
- 5) текущего этажа

10. Пример работы программы (фрагмент)

```
1 Выберите
2 расу:
3 1 - Человек
4 2 - Эльф
5 3 - Дворф
6 > 2Ваш
7
8 персонаж создан!
9 HP: 85
10 ATK: 12
11 DEF: 6
12 AFI: 14Вы входите в
13 подземелье...Перед вами
14 развилка.
15 (1) Слева: ???
16 (2) Справа: Комната отдыха Куда
17 пойти?
```

Рисунок 1 – Листинг программы примера работы игры

11. Ограничения и требования

- 1) Программа **консольная**
- 2) Управление через текстовое меню и ввод команд
- 3) Язык программирования - не ограничен <s>(в том числе можно Python)</s>
- 4) Код должен быть читаемым и логически структурированным, можно делить на разные файлы

Содержание

1	Введение.....	9
2	Создание персонажа	9
2.1	Выбор расы	9
2.2	Характеристики персонажа	10
3	Опыт и уровни.....	10
3.1	Прокачка характеристик	11
4	Инвентарь и экипировка	11
4.1	Инвентарь.....	11
4.2	Экипировка	11
5	Подземелье и комнаты	11
5.1	Структура подземелья.....	11
5.2	Типы комнат	12
5.3	Видимость комнат	12
6	Враги и сложность	12
7	Боевая система.....	12
8	Предметы и добыча.....	12
9	Хранение данных.....	13
10	Пример работы программы.....	13
11	Ограничения и требования.....	13
12	Классы.....	13
12.1	Класс Item.....	13
12.2	Класс Weapon.....	14
12.3	Класс Armor.....	14
12.4	Класс Potion.....	14
12.5	Класс Character.....	14
12.6	Класс Enemy	15
13	Основные механики боя	16
13.1	Атака	16
13.2	Использование зелья	16
13.3	Уклонение	16
14	Битва	16
15	Механики вне боя.....	16
15.1	Прокачка уровней.....	16
15.2	Комнаты отдыха.....	16

15.3	Инвентарь и экипировка.....	17
16	Заключение	17

1 Введение

Цель проекта — разработка текстовой RPG-игры. Проект позволил изучить принципы объектно-ориентированного программирования, генерацию случайных событий, пошаговые бои и работу с консольным вводом/выводом. Игра управляется через консоль, игрок создаёт персонажа, исследует подземелье, сражается с врагами и собирает предметы.

2 Создание персонажа

2.1 Выбор расы

В начале игры пользователь выбирает расу персонажа:

- 1) Человек
- 2) Эльф
- 3) Дворф

Каждая раса задаёт диапазоны генерации характеристик.

```
Добро пожаловать в текстовую RPG!
Выберите расу:
1 - Человек, 2 - Эльф, 3 - Дворф
1

Ваш персонаж создан!

Human - Уровень 1
HP: 85/85 ATK: 11 DEF: 7 AGI: 12
Рост: 184 см Вес: 84 кг
Оружие: Нет
Броня: Нет

Этаж 1, комната 1
(1) Слева: chest
(2) Справа: ???
Куда пойти? 1-налево, 2-направо
```

Рисунок 2 – Листинг программы примера работы игры

2.2 Характеристики персонажа

Характеристики генерируются случайным образом при создании персонажа:

- HP – здоровье
- Attack – сила атаки
- Defense – защита
- Agility – ловкость (влияет на уклонение)
- Height – рост
- Weight – вес

Пример диапазонов по расам:

Раса	HP	ATK	DEF	AGI	Рост (см)	Вес (кг)
Человек	80–100	10–15	5–10	10–15	165–190	60–90
Эльф	60–80	12–17	4–8	14–20	160–180	50–75
Дворф	90–120	14–19	8–13	6–11	140–160	70–100

3 Опыт и уровни

- 1) Персонаж получает опыт за победу над врагами.
- 2) При накоплении нужного количества опыта повышается уровень.
- 3) Каждый новый уровень даёт 5 очков характеристик для распределения.

```

Этаж 1, комната 1
(1) Слева: chest
(2) Справа: ???
Куда пойти? 1-налево, 2-направо
2
Вы встретили Тролль!

Ваш HP: 85/85 | Тролль HP: 32
1-Атаковать, 2-Использовать зелье, 3-Уклонение, 4-Инвентарь
1
Вы нанесли 5 урона .
Тролль атакует и наносит 2 урона!

Ваш HP: 83/85 | Тролль HP: 27
1-Атаковать, 2-Использовать зелье, 3-Уклонение, 4-Инвентарь
1
Вы нанесли 5 урона .
Тролль атакует и наносит 2 урона!

Ваш HP: 81/85 | Тролль HP: 22
1-Атаковать, 2-Использовать зелье, 3-Уклонение, 4-Инвентарь
1

```

Рисунок 3 – Листинг программы примера работы игры

3.1 Прокачка характеристик

Игрок может распределять очки вручную между характеристиками:

- HP (+5 HP)
- Attack (+1 ATK)
- Defense (+1 DEF)
- Agility (+1 AGI)

Распределение очков выполняется в комнатах отдыха или при достижении нового уровня.

4 Инвентарь и экипировка

4.1 Инвентарь

Инвентарь хранит предметы:

- 1) Зелья (лечение)
- 2) Монеты
- 3) Оружие
- 4) Броня

Игрок может:

- 1) Просматривать инвентарь
- 2) Использовать предметы
- 3) Экипировать оружие или броню
- 4) Выбрасывать предметы (кроме монет)

4.2 Экипировка

В инвентаре есть отдельные слоты:

- Оружие
- Броня

Экипированные предметы влияют на характеристики персонажа (увеличивают атаку или защиту).

5 Подземелье и комнаты

5.1 Структура подземелья

- Игра начинается в подземелье.
- Подземелье состоит из комнат.
- После каждой комнаты игрок выбирает путь: налево или направо.
- Каждые 5 комнат повышается этаж и сложность врагов.

5.2 Типы комнат

Комнаты генерируются случайно:

1. Боевая комната – бой с врагом
2. Комната отдыха – восстанавливает HP и позволяет распределить очки
3. Комната с сундуком – находка оружия, брони, зелья или монет

5.3 Видимость комнат

Перед выбором направления игрок:

- Иногда видит, что находится дальше
- Иногда неизвестно (???)

6 Враги и сложность

- Враги генерируются случайно с характеристиками (HP, Attack, Defense).
- Сложность возрастает с каждым этажом.
- Враги получают бонусы к характеристикам в зависимости от уровня игрока и этажа.

7 Боевая система

Бой пошаговый:

1. Игрок выбирает действие:
 - Атаковать
 - Использовать зелье
 - Попытаться уклониться
 - Вызвать инвентарь
2. Атака учитывает:
 - Характеристики игрока
 - Экипированное оружие/броню
 - Случайные факторы (критический удар 10%)
3. Враг наносит урон игроку после хода игрока, если не погиб.

8 Предметы и добыча

- Враги и сундуки могут давать оружие, броню, зелья, монеты.
- Игрок может использовать или экипировать предметы сразу.
- При нехватке места игрок решает, что выбросить.

9 Хранение данных

- Состояние игры сохраняется в JSON (savegame.json)
- Хранятся характеристики персонажа, инвентарь и уровень.

10 Пример работы программы

```
Добро пожаловать в текстовую RPG!
Выберите расу:
1 - Человек, 2 - Эльф, 3 - Дворф
1

Ваш персонаж создан!

Human - Уровень 1
HP: 85/85 ATK: 11 DEF: 7 AGI: 12
Рост: 184 см Вес: 84 кг
Оружие: Нет
Броня: Нет

Этаж 1, комната 1
(1) Слева: chest
(2) Справа: ???
Куда пойти? 1-налево, 2-направо
```

Рисунок 4 – Листинг программы примера работы игры

11 Ограничения и требования

- Программа консольная
- Управление через текстовое меню
- Язык программирования – C#
- Код читаемый, структурирован по классам

12 Классы

12.1 Класс Item

Базовый абстрактный класс для всех предметов.
Содержит только название предмета.

```
Ссылок: 7
abstract class Item
{
    public string Name;
}
```

Рисунок 4 – Листинг класса Item

12.2 Класс Weapon

Наследуется от Item, содержит бонус к атаке.

```
class Weapon : Item
{
    public int AttackBonus;
    Ссылок: 1
    public Weapon(string name, int bonus) { Name = name; AttackBonus = bonus; }
}
```

Рисунок 2 – Листинг класса Weapon

12.3 Класс Armor

Наследуется от Item, содержит бонус к защите.

```
class Armor : Item
{
    public int DefenseBonus;
    Ссылок: 1
    public Armor(string name, int bonus) { Name = name; DefenseBonus = bonus; }
}
```

Рисунок 3 – Листинг класса Armor

12.4 Класс Potion

Наследуется от Item, восстанавливает здоровье.

```
class Potion : Item
{
    public int HealAmount;
    Ссылок: 1
    public Potion(string name, int heal) { Name = name; HealAmount = heal; }
}
```

Рисунок 4 – Листинг класса Potion

12.5 Класс Character

Хранит характеристики игрока, инвентарь, экипировку, уровни и очки статов.

Содержит методы:

- GenerateStats() — генерация характеристик по расе
- GainExp() — получение опыта
- LevelUp() — повышение уровня и распределение очков
- ShowStats() — отображение характеристик
- DistributeStatPoints() — распределение очков
- ShowInventory() — просмотр и использование предметов

```

-----+
class Character
{
    public string Name;
    public Race Race;
    public int Level = 1;
    public int Exp = 0;
    public int ExpToLevel = 100;
    public int MaxHP;
    public int HP;
    public int Attack;
    public int Defense;
    public int Agility;
    public int Height;
    public int Weight;
    public int StatPoints = 0;

    public Weapon EquippedWeapon = null;
    public Armor EquippedArmor = null;
    public List<Item> Inventory = new List<Item>();

    private static Random rand = new Random();
}

```

Рисунок 5 – Листинг класса Character

12.6 Класс Enemy

Генерируется случайно с характеристиками в зависимости от уровня.
Примеры врагов: Гоблин, Скелет, Тролль, Разбойник.

```

class Enemy
{
    public string Name;
    public int HP;
    public int Attack;
    public int Defense;
    private static Random rand = new Random();

    Ссылка 1
    public Enemy(int level)
    {
        string[] names = { "Гоблин", "Скелет", "Тролль", "Разбойник" };
        Name = names[rand.Next(names.Length)];
        HP = rand.Next(20, 31) + level * 5;
        Attack = rand.Next(5, 11) + level * 2;
        Defense = rand.Next(2, 6) + level;
    }
}

```

Рисунок 6 – Листинг класса Enemy

13 Основные механики боя

13.1 Атака

- Физическая атака учитывает базовую атаку персонажа + бонус оружия
- Есть 10% шанс критического удара
- Урон уменьшается на броню противника

13.2 Использование зелья

- Восстанавливает HP персонажа
- Зелье исчезает из инвентаря после использования

13.3 Уклонение

- Шанс уклонения = $\text{Agility} * 2 - \text{Attack}$ врага
- Генерация случайного числа для успеха или провала

14 Битва

- 1) Игрок выбирает действие
- 2) Атака или использование зелья
- 3) Враг наносит урон, если жив
- 4) Цикл продолжается пока не победит один из участников
- 5) Игрок получает опыт после победы

```
while (player.HP > 0 && enemy.HP > 0)
{
    Console.WriteLine($"\\nВаш HP: {player.HP}/{player.MaxHP} | {enemy.Name} HP: {enemy.HP}");
    Console.WriteLine("1-Атаковать, 2-Использовать зелье, 3-Уклонение, 4-Инвентарь");
    string action = Console.ReadLine();
    switch (action)
    {
        case "1": Attack(enemy);
        case "2": UsePotion();
        case "3": Dodge();
        case "4": Inventory();
    }
}
```

Рисунок 7 – Фрагмент цикла битвы в методе Battle

15 Механики вне боя

15.1 Прокачка уровней

- Каждые 100 опыта повышается уровень
- При повышении уровня начисляются 5 очков характеристик

15.2 Комнаты отдыха

- Восстанавливают HP
- Позволяют распределять очки характеристик

15.3 Инвентарь и экипировка

- Позволяет использовать зелья
- Экипировать оружие и броню
- Просматривать количество монет
- Ограничений по использованию нет, кроме монет

16 Заключение

Проект позволил реализовать текстовую RPG с большим выбором действий: от создания персонажа до управления его характеристиками. С каждым шагом в игре становится всё интереснее благодаря пропорциональному росту силы противников. Проект также позволил научиться работать с большими объёмами кода, и я с радостью буду ждать схожих заданий.