

Student Name: _____

Student ID: _____

TA: _____

Rust Lab 06 – Text Adventure Game in Rust

6/8/2025

Learning Objectives

- Apply `struct` and `enum` for organizing game entities.
- Use `match` and control flow logic to handle events.
- Manage ownership and references in combat logic.
- Use the `rand` crate for random encounter generation.

Game Overview

You will create a simplified text-based adventure game. The player moves through a map, encounters events, and sometimes fights enemies.

The goal is to collect 100 gold before running out of HP or stamina.

Player Stats:

- HP: 100 (max)
- Stamina: 50
- Power: 10
- Gold: 0

Encounters and Effects:

Encounter	Effect	Chance
Nothing	Nothing happens	25%
Meat	+5 HP (max 100)	20%
Water	+2 stamina	20%
Herb	+1 power	15%
Enemy	Start combat	20%

Enemies:

Enemy	HP	Power	Gold Reward	Encounter Chance
Rat	10	2	10	60%
Wolf	20	5	20	30%
Boar	30	10	30	10%

Tasks:

1. Define struct `Player` and `Enemy`.
2. Define enums `Direction` and `Encounter`.
3. Implement movement and reduce stamina.
4. Generate random encounters using rand crate.
5. Handle effects (meat, water, herb, enemy).
6. Implement simplified combat logic.

7. Loop game until player wins or loses.

Sample Code Snippets

Random Encounter Selection (using rand):

```
use rand::Rng;

let roll = rand::thread_rng().gen_range(0..100);
let encounter = match roll {
    0..=24 => Encounter::Nothing,
    25..=44 => Encounter::Meat,
    45..=64 => Encounter::Water,
    65..=79 => Encounter::Herb,
    _ => Encounter::Enemy,
};
```

Getting Direction Input:

```
use std::io::{self, Write};

let mut input = String::new();
print!("Enter direction (N/S/E/W): ");
io::stdout().flush().unwrap();
io::stdin().read_line(&mut input).expect("Failed to read");
let input = input.trim().to_uppercase();
```

Add the following to Cargo.toml:

```
[dependencies]
rand = "0.8"
```

TA Checklist

- __ Code is reasonably formatted and commented
- __ Used `struct` for player & enemy
- __ Used `enum` for direction & encounter
- __ Used `match` control flow
- __ Correct event effect application
- __ Implemented combat and gold logic
- __ Lose condition: HP <= 0 or Stamina <= 0
- __ Win condition: Player reaches 100+ gold
- __ Uses rand crate correctly and listed under [dependencies] in Cargo.toml

TA Comment: _____

Finished Time: _____