Project 1: Text-based Role-Playing Game

The objective of this project is to design a program using loops and conditional statements.

Requirements:

You are contracted to assist with the development of a text-based RPG. The goal of this game is for a single adventurer to navigate an "endless" dungeon until they are defeated or decide to quit and return to town which ends the game.

Specific requirements for this program:

A variable that keeps track of the player's health, this variable should start at 100 hit points.

Display a menu with the following set of actions:

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!

The game should do the following for each of the menu choices:

For option one, the player fights an Orc and takes a random amount of damage from 0 – 20. The game should also output the message. "The player fights off a creature and takes <random> amount of damage!"

For option two, the player takes a moment to heal and rest after adventuring and heals a random of damage from 5 – 15. The game should output the message, "The player takes a moment to recuperate and heals <random> wounds!"

For option three, the player looks around the current room they are in and obtains a random amount of gold from 20 – 200. The game should output the message, "The player loots the dungeon room for <random> gold pieces!"

For option four, the game terminates and outputs, "The player has finished adventuring and returned to town!"

Additional requirements:

The player cannot exceed 100 health no matter how much they rest.

If the player reaches or drops below 0 health the game ends with the message, "The player has been defeated!"

If the player reaches 2000 gold the game will end with the message, "The player has conquered the dungeon!"

The player can only loot a dungeon room once until they choose option one again. The game should output, "The player has looted this room, please venture forth!"

The game should output, “Incorrect selection, please choose 1-4.” On any incorrect input.

Example outputs:

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!
1
The player fights off a creature and takes 6 amount of damage!

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!
1
The player fights off a creature and takes 16 amount of damage!
The player has been defeated!

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!
2
The player takes a moment to recuperate and heals 5 wounds!

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!
3
The player loots the dungeon room for 190 gold pieces!

How do you wish to continue your adventure?
1. Venture forth!
2. Rest.
3. Check for loot.
4. Retire!
3
The player has looted this room, please venture forth!

How do you wish to continue your adventure?
1. Venture forth!

2. Rest.
3. Check for loot.
4. Retire!
4
The player has finished adventuring and returned to town!

Rubric (each item is worth one point):

- Displays each menu item and get user's input.
- Performs each menu action with expected output.
- Handles invalid user input and continues running program.
- Proper use of conditional statements to accomplish tasks in requirements. (Worth 2 points)
- Program only exits correctly when gold limit is reached, player is defeated, or the player chooses to exit.
- The program is well designed and commented.