**Individual Assignment**

**Due Sunday, April 19 by 11:59 pm**

Welcome to the final individual assignment in ITEC-3860! The goal from these series of individual assignments is to warm up your programming skills for the final project deliverable for your text-based adventure game. In games of this sort, the player wanders around from one location to another, picking up objects, and solving simple puzzles. The program you will create for this assignment is considerably less elaborate than the final project deliverable and it therefore limited in terms of number of rooms, items, monsters etc. Even so, you can still write a program that captures much of the spirit and flavor of the final game.

This handout contains what you need to know about the final individual deliverable.

**Assumptions:**

1. Player is able to navigate between rooms
2. Map information (including rooms’ descriptions, connections) are being retrieved from text file
3. Your code is keeping track of visited rooms
4. Player is able to examine an item, pick up an item and drop off item in any chosen room.
5. Player is able to examine puzzle, solve and/or ignore a puzzle.
6. Player is able to access his/her inventory at any time
7. You coder keeps track of player inventory (update inventory based on picked up or dropped items)
8. Your code keeps track of number of attempts for solving a puzzle.
9. Rooms, Items and puzzles information are being retrieved from text file

**Goal:** Expand the Item feature and implement the monster feature as directed below. All monster/item information must follow the pattern of being retrieved from a text file.

**Expansion of Item feature:**

For this feature your goal is to expand the behaviuor of the item feature (from 2nd individual assignment) to include equip, un-equip and heal commands as described below:

Note: the commands below are applicable to the player only (not monster) and can be used before engaging in a fight with a monster or as the player is actively engaged in a fight.

* **Equip ITEM-NAME: this command will result in increasing the attack damage of the player based on the attack points assigned to the chosen item.**
* **Unequip: this command will result in retrieving the original state of the player attack damage if any were assigned to the player who carries no fighting items.**
* **Heal: using this item will result in increasing the player health points based on the assigned HP to the item.**

**Monster Feature:**

Your goal is to allow the player to interact with one monster of your choice in any of the rooms. The Interaction behaviuor with the monster should include the following commands:

1. **Examine Monster:** this command will allow the player to retrieve the description of the examined monster. Your game should display the description of the examined monster to the console/GUI. The monster discerption will include a **written description** of the monster and the amount of its **attack damage**. The monster attack damage will be determined based on the following criteria:
   1. The monster will deal double its attack damage if the randomly generated number at the time of initiating the battel falls below the monster threshold.
   2. Otherwise, the monster will deal its predetermined attack damage.
2. The player will be given the following choices after examining a monster:
   1. **Attack:** if the player chooses to attack the monster, the player will be locked in the fight/combat mode. During the fight mode the player should have access to the inventory and should be able to examine items, equip, and un-equip.
   2. **Ignore:** This will retrieve the original status of the game before encountering the monster and **the monster should disappear from the room and never show up again if the player navigates back to the same room.**
3. **Attack**: this command will move the player to the combat mode to al**low the player to engage in a fight with the monster:**
   1. If the player has no items equipped, the player will deal damage based on its assigned deal damage points.
   2. If the player chooses to equip an item before or as being activity engaged in a fight, the player will deal damage based on the assigned deal damage points to the item of choice.
   3. **As part of the combat system there should be a method that let the player and monster battle against each other until one of them dies. In each player turn, the method should display the player hit-points (life) and monster hit-points (life), then ask the player for an attack or heal command, and the method should check if monster is killed. In each monster turn, the method let the monster attack the player and checks if player is killed.**
   4. **The player wins the battel if he/she is able to kill the monster e.g. monster hit-points = 0.**
      1. **If the player wins, a winning message should be displayed on the console/GUI to inform the player of his/her win. In addition, the monster should disappear from the room and never show up again if the player navigate back to the same room.**
      2. **The player should stay in the room where the fight took place and should be able to continue playing the rest of the game.**
4. **If the player lose the game, a game over message will be displayed to the player and the player must be given two choices**
   1. **Exit: this command will exit the player of the console/GUI**
   2. **Start new game: this will allow the player to restart the game from the beginning.**

**Important notes:**

1. Don’t send me screen shots of your code and don’t ask me to trouble shoot your code. This is not a programming class but I found over the years of teaching this class that students need similar programming to review their programming skills and to avoid unnecessary drama toward the end of the semester. To help you I will discuss possible solutions in the class and show you sample code and you will put time and efforts to get your code up and running properly.
2. Use either JAVA Eclipse or IntelliJ
3. Grading code takes lots of time so keep your code clean, organized and understandable by adding comments. The more organized your code is the quicker I can grade and the faster you will get a feedback.
4. Don’t fix the file path in your code
5. If your code doesn’t run you will get ‘0’ for this assignment. If your code runs but have partial behaviuor you will get partial credits. e.g. your code runs and allows me to navigate between rooms but doesn’t keep track of visited rooms, then I will take points off for this missing requirements.

**What you need to submit:**

1. Export your Java project folder, zip it and submit it on d2l. Don’t submit individual classes.
2. README file. The readme file should include explanation of each text file data structure. Clear map with room labels, connections, items and puzzles locations.
3. Clear screen shoots of given scenarios to show how to navigate, interact with items, interact with puzzle, and interaction with monster.
4. Clear screen shoots of the combat mode while the player is actively engaged in a fight with the monster.
5. Failing to follow the steps above 1 to 5 will result in “0”.