

COMP1006/1406 – Fall 2017

Each team will submit a single file called `adventure-XXX.zip`, where XXX is your team name, to cuLearn.

The assignment is out of 100 marks.

Assignment 4, Assignment 5 and the Project are essentially combined into a single assignment. You will work with your Team and submit a single solution to cuLearn and then demo your solution in the last week of tutorials.

Throughout this assignment specification, N will denote the number of members in your team for the project. This number will generally be 4 or 3.

Deliverables

- i) During your tutorial in the week of November 13-17, your team will submit to one of the TAs in the tutorial your team name and all team members (name and ID).
- ii) On Monday, December 4, each team will have submitted a single zip file with your team's submission. The due date is 7:00PM on that day.
- iii) During your tutorial in the week of December 4-8, your team will present (demo) your code.

Grading

It is assumed that each team member will make a meaningful contribution to the overall submission (and demo). Each member should be contributing *roughly* the same amount of work. If it is deemed that all members contributed about the same, each member of the team will receive the same grade for the assignment and project.

If it is determined that some members did not contribute to the solution then their grade will be individually determined.

Keep in mind that this is a *group* assignment and project. A student that does not contribute to the group effort hurts the group. Equally as damaging is a student that dominates the project and places unreasonable demands on the rest of the team members.

To help in assessing the group nature of the assignment/project, each team member will be individually submitting a team report. Details of this will be posted later.

1: The Game

Here is a brief description of the game. A (human) player will enter the game world (that already contains other players and things). The goal of the game is for the (human) player to retrieve something from the game world and exit with it (it might be another player or a *thing*).

The game world will be read from a file that specifies each room in the world. Each room will have a list of contents (things), players, and exits (doors to other rooms).

The details of the game are left to your team.

2: Details

Your game must have the following classes: **World**, **Room**, **Player** and **Thing**.

World: Your world class will need to read the game world from a file. It will need to keep track of the entrance to the world so that the human player can enter it. Your world class should also be able to *save* a game in progress. That is, save the state of the game so that it can be loaded later and continued (from the same point in the game). Your world will need to hold your collection of rooms in some way. (You can use either an array or a linked-structure.)

Room: A room is where everything happens. A room contains players, things and doors to other rooms. A player should be able to take things from a room and leave things in a room. Doorways between rooms can be simple doors (that can always be opened), be locked (requiring a key to be found first) or one-way (you can go from room A to B, but not from B to A). A basic room will have 1-4 adjacent rooms. You do not need to consider a room having more than one doorway (adjacent room) in any given direction.

Player: Each players should have a specific role in the game. There will be several players in the game including a human players and at least N more players. Each team member will contribute a specific computer controlled player that have different behaviours.

For the human player, we will North, South, East, and West to denote directions. N, S, E and W will be sufficient.

Thing: There will be several things in the game. Examples might food, treasure, radioactive material, keys, locked boxes, etc. There needs to be at least N different things that have different behaviour.

A good solution will have at least two abstract classes, several concrete classes and possibly an interface or more.

3: File Format

The file format for the world will be as follows:

```
total number of rooms
room number
room name
rooms adjacent to room (comma separated)
players in room (comma separated)
things in the room (comma sepatated)
room number
room name
rooms adjacent to room (comma separated)
players in room (comma separated)
things in the room (comma sepatated)
...
```

For example, if there are 4 rooms in the world (2x2 grid) then the file might look like (# denotes a comment and will not be part of the actual file; I have included it for clarity of the file)

```

4          # 4 rooms in total
1          # room 1
Entrance   # room 1 is the Entrance
2, 3       # you can go to room 2 and 3 from room 1
0, 1, 0, 1 # there is one player1 and one player3 in the room
1, 1, 0, 0, 0 # there is one thing0 and thing1 in the room
2          # room 2
CS Lab     # room 2 is the CS lab
1, 4       # you can go to room 1 and 4 from room 2
0, 0, 0, 0 # there are no players in the room
0, 1, 0, 0, 0 # there is one thing1 in the room
3          # room 3
Washroom   # The washroom
1, 4       # you can go to room 1 and 4 from room 3
0, 0, 0, 0 # no player is in the washroom
0, 0, 0, 0, 1 # there is one thing4 in the room
4          # room 4
2, 3       # you can go to room 2 and 3 from room 4
1, 0, 0, 0 # there is one player1 in the room
0, 0, 1, 0, 1 # there is one thing2 and thing4 in the room

```

For the players and things, I assume you have 4 different player classes and call them for simplicity player0, player1, player2, player3 in the comments above. In your program, you will use your own classes. Same for the things. Here, thing2 is Rodney is found in room 4 of the game.

You can store the game using a 2d array or using a linked structure. Any file that we give you will be given as above and you can assume that the world looks like a rectangle. The first room (#1) is the top-left room. The rooms are then given (in order of room number) and will be given as the rows. For example, the world of the file above corresponds to a world that looks like

1	2
3	4

You will need to determine the shape (number of rows and columns) based on which rooms room # 1 is adjacent to.

4: Example

Here is an example that you can use for your assignment/project. [Note: no disrespect is intended for any student of the University of Ottawa or any students in any discipline. If you find that it is disrespectful, please contact me personally so that I am aware of this and not use the same example in the future.]

Rodney the Raven has gone missing. It is suspected that he has been abducted by some nefarious UO students and being held captive somewhere on the UO campus. Your (human player) task is to explore the UO campus to find and bring Rodney back to Carleton.

Being a student, you are always hungry and on the look out for some food. If you go without food for too long, you'll fall asleep and fail your mission. You'll also be on the lookout for power outlets to charge your phone. Without your phone, you won't be able to call for a ride back to Carleton once you have rescued Rodney.

Along the way, you'll encounter several UO students: a humanities student will give you poutine to eat but then steal your money while your are eating it to give to charity; a physics student will

randomly walk around the hallways of school, slightly mad from playing with too much mercury, and offer you some food (the food might be radioactive though and will continue to hurt you remove health; well after you have eaten it; but it might also be good); a business student will sell you a hint to find Rodney; nefarious students who have been involved with Rodney's abduction. These students Will go and move Rodney once they see you (and know someone is looking for him) or go an empty vending machines so that you cannot buy food; and computer science students that will pay you money help them with their assignments. Any student might ask to use your phone in exchange for food. Some will do this to drain your power (without giving you any food) and others are honest.

In the rooms, you might find a power outlet to charge your phone, vending machines (that sell food that might be good or bad), free food, or fidget spinners. If you take a fidget spinner, it will occupy all your time so that you don't eat and all your health will go away until you fall asleep.

5: More to come

A rubric for the assignment and project will be posted in the coming days. Be sure to ask questions on the class forum when you are not certain of what to do.

Remember that this is a group effort. You will need to find a good way of collaborating and sharing your code. Your team has a significant amount of freedom to choose what you want to do. Be sure to meet (or talk) early and get started early.

I will post some specific things that you can add to your program if you looking for more ideas. Individuals can do *extra* work to offset some lower grades for an earlier assignment.