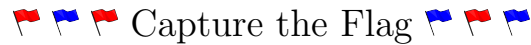


COMP1006/1406 – Summer 2016

Submit a single file called `assignment8.zip` to cuLearn.



Capture the Flag

In this assignment you will build on your players from the last assignment to implement the game Capture the Flag. If you are unfamiliar with the game, you can read about it here: http://en.wikipedia.org/wiki/Capture_the_flag.

There are many variants of the game (as you can see from the wiki document). For our game, the rules will be defined below in this project document.

1: The Basic Game Rules

Capture the flag is a game played by two teams, which we'll refer to as team 1 (**Blue** team) and team 2 (**Red** team), on a playing field which is divided into two territories.

Each team in the game has

- a “territory” on the playing field (side 1 or side 2),
- a “base” that is located somewhere in their territory,
- a “jail” that is located somewhere in their territory,
- six or more players, and
- a “flag” (that is initially located at their base).

The object of each team is to capture the opposing team's flag and bring it back to their base. The first team to do this wins the game.

There are several notions of being “caught” in the game that depend on where a player is and if they have the flag or not. Being caught essentially means an opposing player is “close enough” to them to catch them.

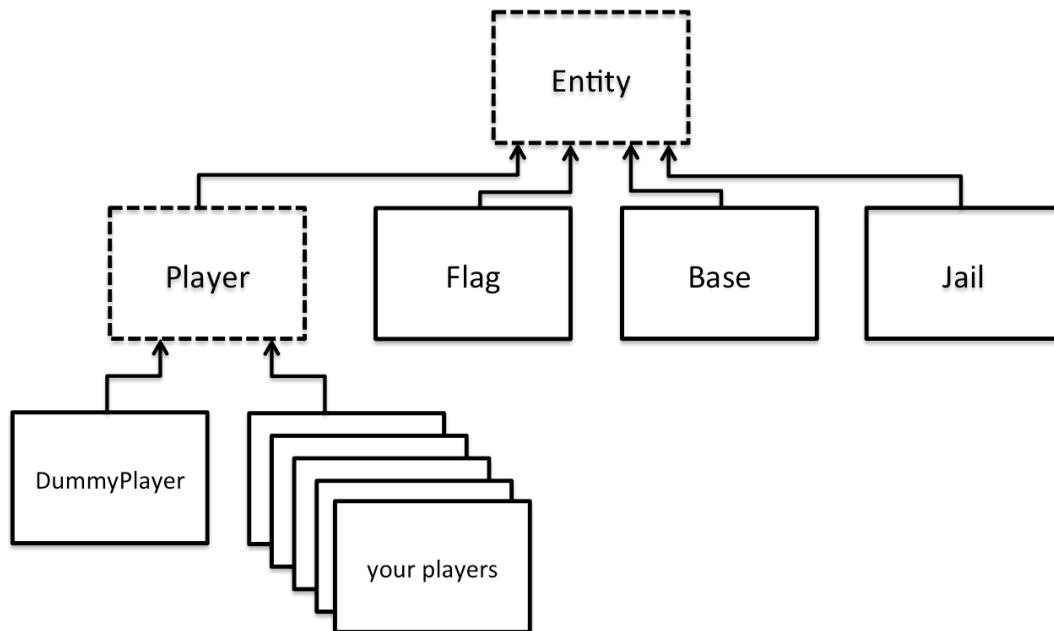
- A player can be caught when they are on the opposing team's territory. When they are caught, they are brought to the opposing team's jail (by the player that caught them). They remain in jail until they are freed by a member of their own team (that is not in jail). If the player was carrying a flag they will drop the flag where they are caught.
- A player can be caught on their own territory only if they are also carrying the opposing team's flag. When they are caught, they will first drop the flag where they are and then be brought back to the opposing team's jail (by the player that caught them). They remain in jail until they are freed by a member of their own team (that is not in jail).
- When a player is in their own territory and is not holding the other team's flag they cannot be caught.

When a player is caught and sent to jail they will remain in jail until another player on their team, who is not in or being taken to jail, “frees” them. A player frees a teammate by being “close enough” to them.

2: Classes in the Game

There are several classes involved in the game. You should already be familiar with these classes. Some are provided and should not be changed (alter at your own risk!), some are provided and need to be completed, and some you will need to create.

The “things” in the game all extend an abstract **Entity** class.



Entity: This abstract class provides the base for “things” in the game such as players, flags, jails and bases.

Player: A direct child of Entity, this abstract class contains the basic framework for all players in the game.

DummyPlayer: A provided concrete player that doesn’t do much. It is used to get the game up and running as a starting point.

Your own Players: You will implement several different kinds of Players that behave differently in the game.

Flag: A direct child of Entity, this is for the two flags in the game. They will not do much in a basic game (just keep a location).

Jail: A direct child of Entity, this is for the two jails in the game. They will not do much in the basic game (just keep a location).

Base: A direct child of Entity, this for the two the bases in the game. They will not do much in the basic game (just keep a location).

Team: You will create a **Team** class that holds a team of players. (this is a new class)

Field: This class provides the base for the playing field and for overseeing the actual game.

View, SpriteStore, Sprite: These classes provide graphics for the game. You do not need to modify these.

CaptureTheFlag: This is the game itself. It drives the game (with a `main` method).

3: The Players

In a game of capture the flag each team will have six or more players. You will implement four different behaviours for them. In particular, you will need to implement the following.

- Players that try to capture the opponents flag and bring it back to their home (to win)
- Players that try to catch opponents (so that they go to jail).
This is the only player that is allowed to catch other players **.
- Players that try to free their teammates from jail
- Players that do something different than the above three.
(** this player might also be able to catch other players)

All the players must stay in the playing field at all times.

4: The Team Class

Write a `Team` class that has a constructor

```
public Team(Field theField, int numFlagGetters, int numCatchers,  
            int numFreeers,      int numYourPlayer)
```

You can add other constructors if you wish but this one must be present. The purpose of the `Team` class is, as the name suggests, to represent teams in your game. The intention is that your `CaptureTheFlag` game will have code similar to

```
Team reds    = New Team(field, 1, 2, 3, 4);  
Team blues   = New Team(field, 3, 1, 1, 2);
```

which creates, initializes and registers all players that will be playing in the game. The reds team will have one player that tries to capture the flag, two players that try to capture opponents, three players that try to free teammates in jail, and four players that you define yourself. This will replace the `for` loop that creates players in the `main` method of the provided `CaptureTheFlag.java` file.

5: The Field Class

Finish the incomplete methods in the `Field` class so that your players can play the game. In particular, you should write code for or modify the existing code in

- `winGame`
- `freeTeammate`
- `pickUpFlag`
- `catchOpponent` This code is not robust but works.

There are several (empty) methods in the `Player` class that you can implement to help you with your implementation of the Field methods. (The methods should provide some functionality to allow a player to know, for example, if it has the flag or if it has been caught. Add any other methods, like `isInJail` or `hasFlag`, as you need.)

Submission

In a plain text file called `a8.txt`, briefly describe what your 4th player does in the game. If you add any new configuration values to the Team constructor also list the values and give a brief description of the team it creates.

Put all of your java files and `a8.txt` in a file called `assignment8.zip` and submit to cuLearn.