# Game Development 1B – ICE Test 1

## Assignment Brief

You must create a battle simulation between two objects. You must do this alone, with no help from your fellow peers. You may ask your lecturer for help, or utilise the internet. General test conditions otherwise apply.

## Instructions

1. Create a new console application.
2. Create two classes. One for the Hero, and one for the Enemy. They should have relevant class variables:
   1. Name
   2. HP
   3. Damage
   4. Critical Chance
   5. Enemy has its own additional variable: Type
3. Create constructors that initialise the Hero and Enemy sensibly.
4. Create a method for the Hero class called “elvenName()” that receives this name and returns a string of that name with the characters in alphabetical order (“Mhairi” becomes “Ahiimr”)
5. Create randomiseHP(), randomiseDamage() and randomiseCrit(). Each of these methods take a minimum and maximum value and returns a rolled value between these two extremes. Set the relevant variables for the Hero by calling each of these methods in the constructor. Provide relevant values for each extreme.
6. Create rollEnemyType() in the Enemy class. This returns a number to indicate the type of enemy the Hero will be facing:
   1. 1: Goblin
   2. 2: Dragon
   3. 3: Ugandan Knuckles
7. You should set the enemy name in the Enemy constructor based off to results of the roll from the above method. Use a switch statement to return the relevant name (“Goblin” etc.)
8. Create the randomise functions from Hero in Enemy as well, as call them in the constructor to you created to randomise HP, damage and critical chance for the enemy.
9. Create MyString() methods for the Hero and Enemy classes that outputs a string that represents a stat sheet in the console as seen below:

Hero Name:

Hero HP:

Hero Damage:

Hero Critical Chance:

Enemy Name:

Enemy HP:

Enemy Damage:

Enemy Critical Chance:

1. Create a critCheck() function for each class. It should receive a critical chance percentage, roll to see if a critical hit “lands” and return a true or false (e.g. If a critical chance is 25%, and you roll a random number from 1-100, a number from 1-25 represents a successful critical hit).
2. Create an attack() method. It should receive the target as an object and damage that target.
3. Once all this is done, you should be able to create a basic game loop with a while loop (while they’re both still alive) where Hero Attacks, then Enemy Attacks etc. Each turn, check for a critical hit, and if so, double the hero damage output for that turn (pass 2 x attack as the attack parameter to attack())
4. Print out to the console the results of each turn:

e.g.

The fight has begun!

Our Hero, Ahiimr, attacks the Ugandan Knuckles

Ahiimr scores a critical hit!

Ahiimr does 84 damage to Ugandan Knuckles

Ugandan Knuckles has 15 HP left

Ugandan Knuckles attacks Ahiimr

Ugandan Knuckles does 16 damage to Ahiimr

Ahiimr has 156 HP left

1. These values will be randomised and change from run to run. Output some text when the fight ends, declaring the victor.
2. Once your program is complete and working, think about the values you chose for minimum and maximum randomisation elements, and tweak them if one party is winning too much.
3. Show the program to your lecturer and explain your choices.

## Submission

Show the program to your lecturer when it is complete to get marked. Be sure to do this before the end of class, or you will get no marks.