# Program Reflection

I designed the base class ‘Character’ to hold the fields and members that all other classes would contain, giving it a name and stat points. This means that the Item class, the Player class and Monster class all have stats for attack, defence, health etc. attached to them.

I first encountered a lot of trouble with organising the attack from the player and the monster, until I recognised the important significance of private data when it comes to debugging. So I passed the methods to do this into the respective classes and used pointer arguments to manipulate the battle from within the class itself.

With this, I heavily implemented strict private data types doing almost all manipulations of class data from internal voids and functions. This proved far more efficient and made a significant impact on debugging the program since the source of error could be easily identified on the class level.

And a change I would implement is to complete more class methods with more pointers and even use the vector in the player class of the inventory items with pointers, therefore reducing possibility of errors when adding and consuming items. There were still 2 variables in the player class that were made public for the sake of simplicity in only ‘reading’ from their methods, as temporary duplicates made to read from would be more complicated, but definitely will improve the quality of the program being developed.