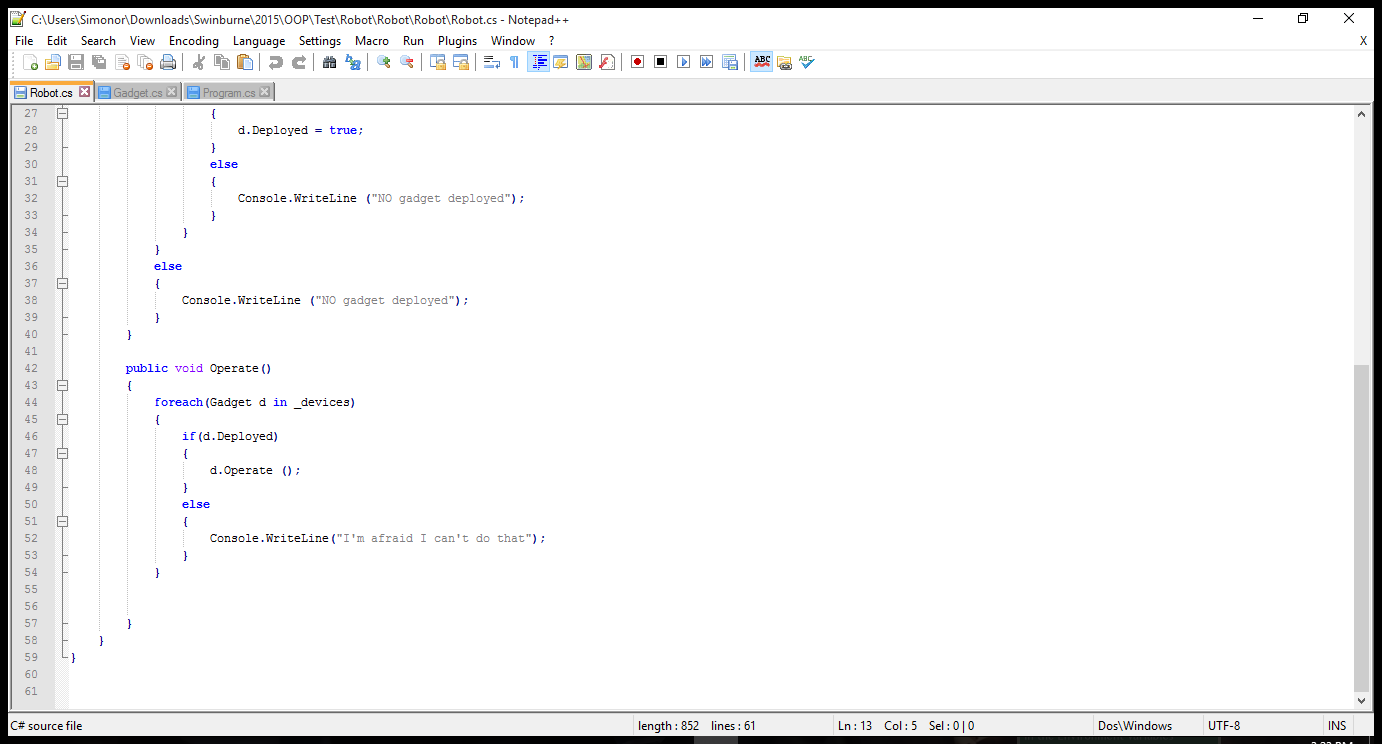
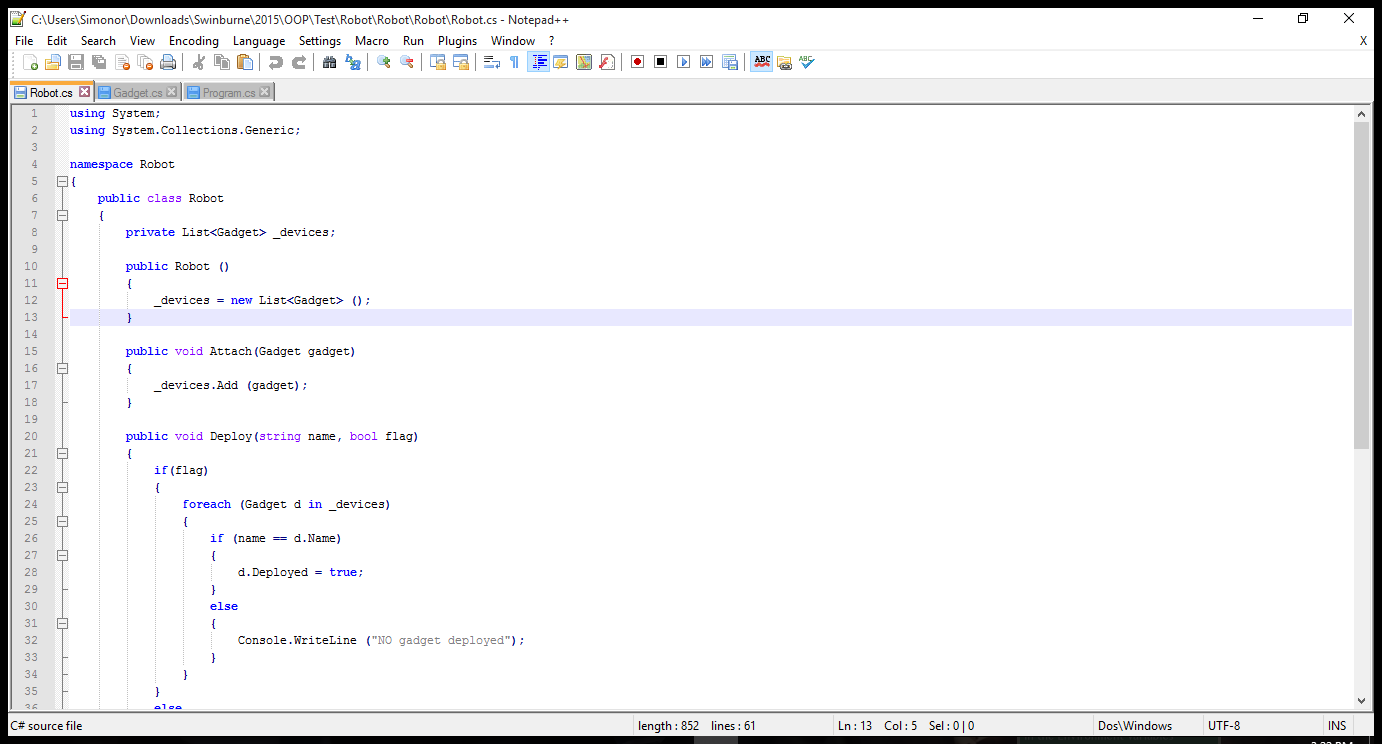
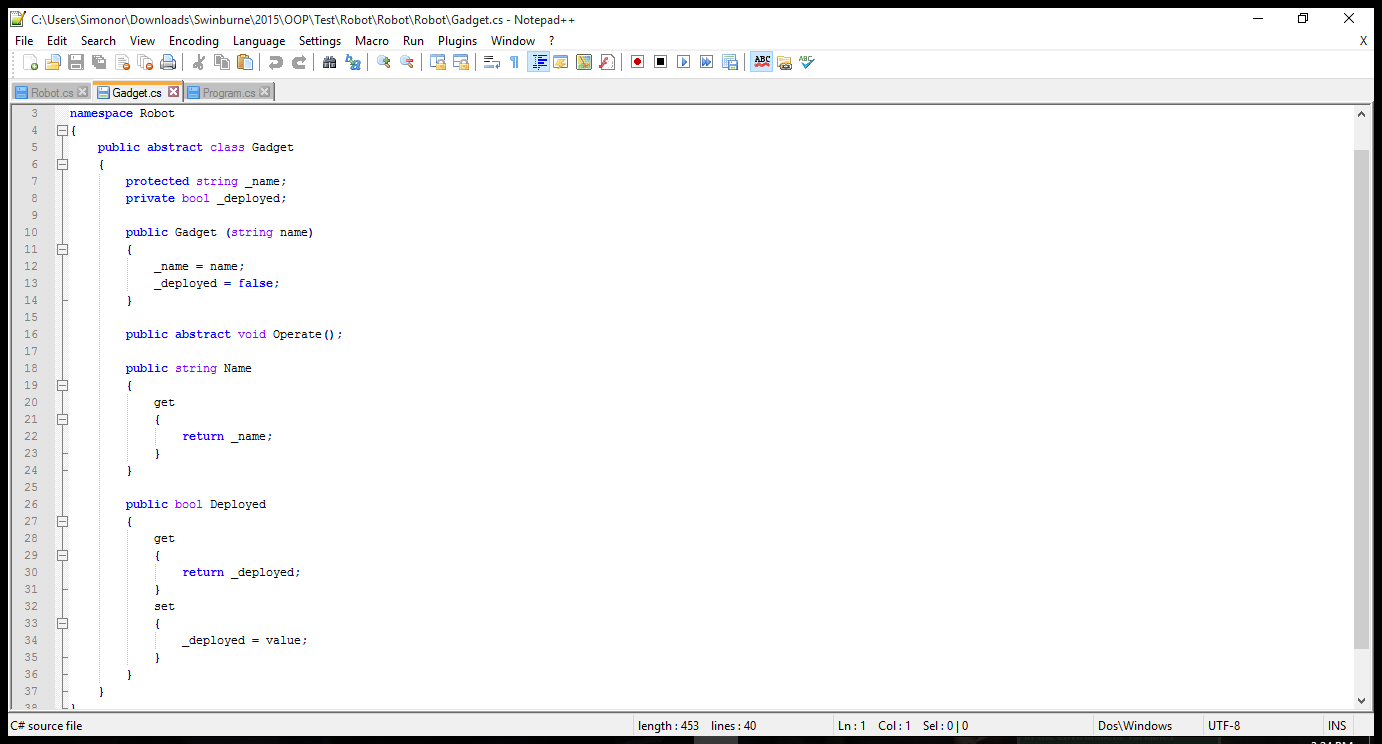


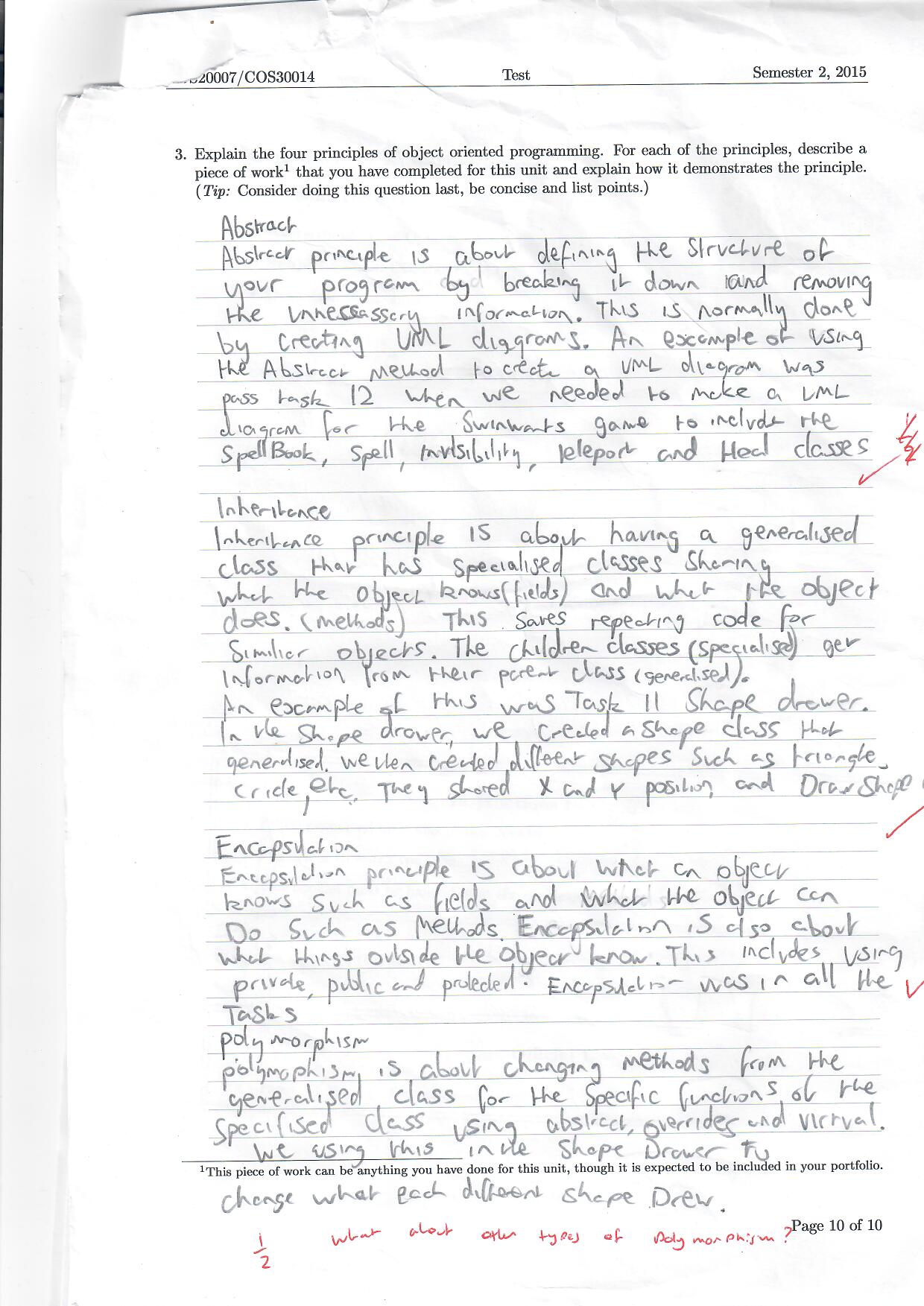
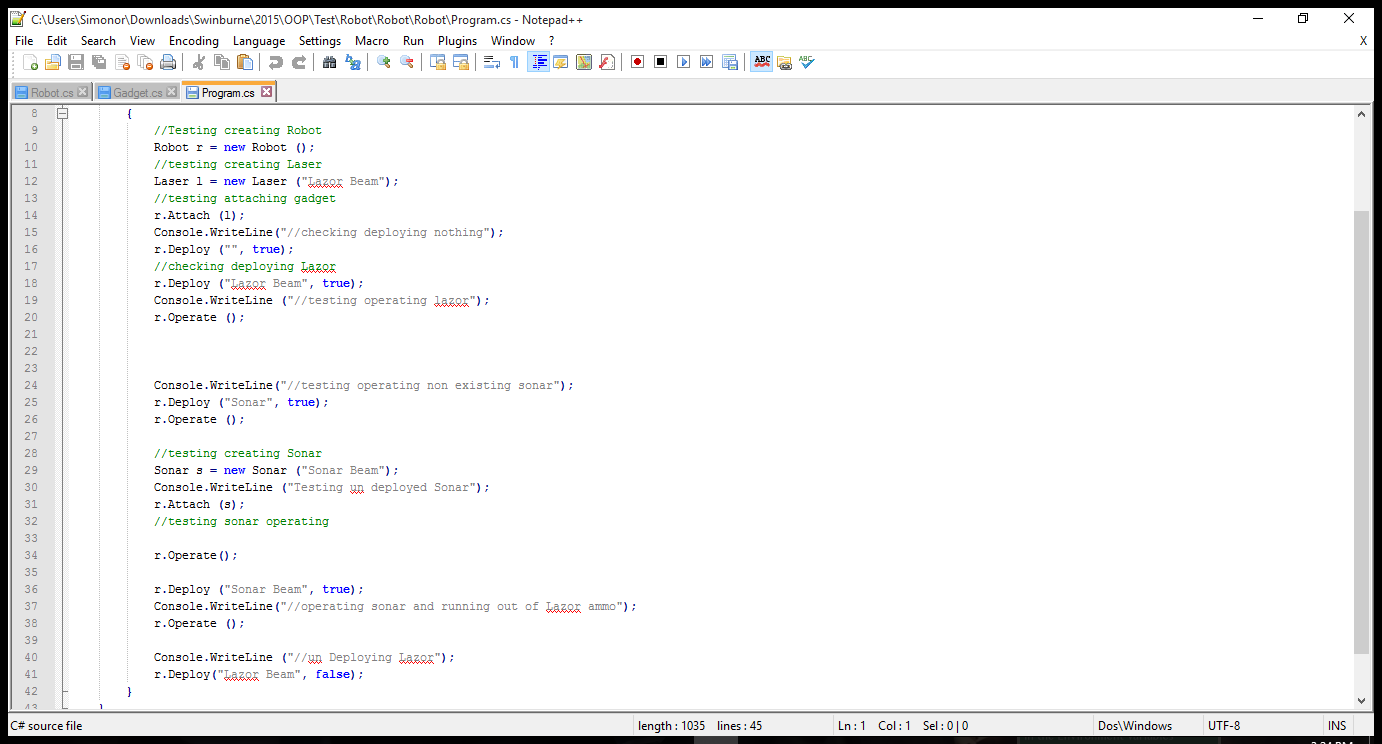
Robot.cs



Gadget.cs



Testing



Polymorphism

polymorphism is the process of child or specialised classes possessing methods from the parent/Generalised class and modifying them to perform different behaviours. Polymorphism heavily depends on inheritance as classes need to inherit the methods to be able to change them. There are 4 different types of polymorphism. Subtype, Adhoc, Parametric and Coercion. Subtype is the main type of polymorphism as it is using derived classes and base classes to change methods. Parametric gives a way to execute code for different types. adhoc allows you to use methods to act differently for different types. Coercion is used when an object is cast into another object.

An example of using polymorphism in the tasks was the Shape Drawer Program. In the shape drawer program each shape required to be drawn so we created a DrawShape method in the parent class. Polymorphism was needed to make the class abstract so we could overrule the method in the child classes so they could each perform SwinGame’s unique drawcircle, drawrectangle and drawline methods.