4. **Classes**

(a) Write an abstract class called `ClassOne` that contains two methods, `methodA` and `methodB`, one of which is implemented and one of which is not. [3]

(b) Write a concrete class called `ClassTwo` that extends `ClassOne`. Do not use method overriding in your implementation. [2]

(c) Write an implementation of `ClassTwo` that does use method overriding. [2]

(d) Write an interface called `InterfaceOne` that contains type signatures for two methods called `methodC` and `methodD`. [2]

(e) Now write a new concrete class called `ClassThree` that is based on `ClassOne` and `InterfaceOne`. [4]

(f) Explain the term multiple inheritance. You should use the definitions above to illustrate your answer. [2]

(g) Using the following class definition

```
public class C  {
   public  static int a;
   private static int b;
   public  int x;
   private int y;
}
```

explain how you would access the values of `a`, `b`, `x` and `y` from another class. If further code is needed in class `C` then you should document this, being careful to explain the selection of any modifiers that are used. [5]