Further Java

(a) Describe the differences and similarities between abstract classes and interfaces in Java. How would you select which kind of definition to use? [5 marks]

(b) An enthusiast for programming with closures proposes extending Java so that the following method definition would be valid:

```java
Closure myCounter (int start) {
    int counter = start;
    return {
        System.out.println (counter ++);
    }
}
```

The programmer intends that no output would be produced when this method is executed, but that it would return an object of a new type, Closure, and that invoking apply() on that object would cause successive counter values to be printed.

By using an inner class definition, show how this example could be re-written as a valid Java program. [5 marks]

(c) A common programming mistake is to try to define a class to have more than one superclass. For example a naive programmer may write

```java
class AutoTree extends Thread, BinaryTree {
...
}
```

when defining a new kind of data structure which uses an additional thread to keep the tree balanced. Describe three ways in which this problem can be resolved to produce (one or more) valid class definitions. State, with a brief justification, which you would use here. [10 marks]