Programming Language Concepts, CS2104
Tutorial 9 (4 November 2011)
(All students must prepare/attempt in advance.)

Exercise 1

Use the translation scheme given in class to translate the following Java code into C.

```
class Exception1 extends RuntimeException {
    public Exception1(String s) {
        super(s) ;
    }
}

class Exception2 extends RuntimeException {
    public Exception2(String s) {
        super(s) ;
    }
}

class Exception3 extends RuntimeException {
    public Exception3(String s) {
        super(s) ;
    }
}

public class ExceptionExample {
    public static void main ( String [] args ) {
       try {
            first(Integer.parseInt(args[0]),Integer.parseInt(args[1])) ;
            System.out.println("This might not get printed") ;
       } catch (Exception1 e) {
            System.out.println(e) ;
       } catch (Exception2 e) {
            System.out.println(e) ;
       }
    }

    public static void first (int a, int b) {
       try {
            if ( a == 1 )
              throw new Exception1("Exception 1 from first") ;
            second(b) ;
       } catch (Exception3 e) {
            System.out.println(e) ;
       } finally {
            System.out.println("In first");
       }
       System.out.println("In first: "+
            "this might not always get printed") ;
    }
```

```
    public static void second (int b) {
       if ( b == 1 )
           throw new Exception2("Exception 2 from second") ;
       if ( b == 2 )
           throw new Exception3("Exception 3 from second") ;
       System.out.println("In second") ;
    }
}
```

Exercise 2

Using the translation scheme for object oriented programs given in class, translate the following Java code into C.

```
class Drawable {
    protected int scaleFactor = 1 ;
    public void draw() {
        System.out.println("Generic drawable object") ;
    }

    public void scaledDraw(int factor) {
        scaleFactor = factor ;
        draw() ;
        scaleFactor = 1 ;
    }

    public static void main(String [] argv) {
        Drawable x ;
        if ( argv[0].equals("c") ) {
            x = new Circle(10,10,10) ;
        } else {
            x = new Square(5,5,5) ;
        }
        x.scaleDraw(10) ;
    }
}

class Circle extends Drawable {
    int x, y, radius ;
    Circle(int x, int y, int radius) {
        this.x = x ;
        this.y = y ;
        this.radius = radius ;
    }
    public void draw() {
        System.out.println("Circle with center at ("
                           +x+","+y+") and radius "+factor*radius) ;
    }
}
```

```
class Square extends Drawable {
    int x, y, side ;
    Square(int x, int y, int side) {
        this.x = x ;
        this.y = y ;
        this.side = side ;
    }
    public void draw() {
        System.out.println("Square with corner at ("
                        +x+","+y+") and side "+factor*side) ;
    }
}
```