Worksheet Three

"Oh that was easy", says Man, and for an encore goes on to prove that black is white and gets himself killed on the next zebra crossing.
Douglas Adams, The Hitch Hikers Guide to the Galaxy 1979

Unit Learning Outcomes Addressed by this worksheet: 1 & 2
For all of the exercises below make electronic copies of all of your solutions. Place these in your P04 directory.

Exercise One.
A man has to take a goat, a tiger and a sack of feed from one city to another. He comes to a river. The only way across the river is by using a small boat which is capable of carrying himself and one other thing (i.e. the goat or the tiger or the sack of feed). If the tiger is left alone with the goat it will kill and eat the goat. If the goat is left alone with the sack of feed it will eat the feed.

In our case the man is a robot. The low level software for controlling the robot has already been developed. There is a class called Robot which has the following sub modules in it that you may use

pickUp: IMPORT thing EXPORT None: robot will identify and pick up the object described by thing (e.g. tiger, goat or feed).
putDown: Puts down whatever its holding. Should never be called if the robot isn't holding anything.
boardBoat: robot will climb into boat. If it is holding something then it will board the boat while holding the item.
sitDown: Robot will sit down. The robot must be in the boat, not holding anything and standing when this sub module is called. The robot will always sit down in the boat such that it is facing the river.
getUp: Robot will stand up. The robot must be in the boat and sitting when this submodule is called.
rowBoat: The robot will row the boat across the river. The robot must be sitting in the boat when this sub module is called. The boat is uni-directional meaning it doesn't need to be turned around.
exitBoat: The robot will climb out of the boat. The robot must be standing up in the boat before this method is called. If the robot is holding something it will be able to climb out of the boat while holding onto the item.

Design an algorithm which allows the robot to successfully cross the river with all three things. You should think about having to defend redundant steps as well as making good use of sub modules to avoid repetition.

Exercise Two
Write Java boolean expressions for the following:

Deciding if xVal (declared as an int) contains an even number.

Deciding if letter (declared as char) is an uppercase letter.

Deciding if x and y (both declared as double) contain equal values.
What other information is required here?

Deciding if year (declared as an int) is a leap year given that a year is a leap year if:
it is divisible by 4
it is not divisible by 100 except where it is also divisible by 400.

Exercise Three.
Design in pseudo code, an algorithm which will:
Input the length and width of a wall as a real number expressing each dimension as a measurement in metres.
Input the area in square metres which can be covered by 1 x 10 litre paint tin.
Calculates the number of tins of paint required.
Outputs the number of tins of paint required for 2 coats of paint.

Exercise Four.
Implement your pseudo code algorithm in Java.

Exercise Five.
A bricklayer has been asked to make the walls of an outdoor bench. The bench is rectangular and will be built on level ground. For the purposes of this algorithm you may assume that each wall is built independently of the other three. At the end of each row the bricklayer must cut the last brick to make the wall the correct length. The other half of the brick is discarded and not used in the wall. You may assume that the height is always in increments of brick height. Design an algorithm in pseudo code which will:

Input the length and height of the front and side walls in meters, as well as the length and height of the bricks to be used.
Calculate how many bricks are required to construct the front and back walls.
Calculate how many bricks are required to construct the side walls.
Calculate and output the total number of bricks required.

Exercise Six.
Translate your algorithm into a Java application.