SCC110: Software Development Term 1. C Programming.

Problem Set 5: Quiz & Creating Your Own Functions

The Goal
This week's lab aims to:

- Test your current understanding of see to identify where you may need to improve your knowledge
- Practice declaring and calling your own functions in today's practical exercise

Part 1: Assessed Quiz
The aim of the first part of today's lab is to undertake **the assessed quiz** on moodle.
Our advice would be:

- Ensure you read the questions very carefully, the code answers in particular assume your program produces exactly the output specified, so take extra care with spaces, newlines etc. *The questions testing your ability to read and interpret code assume you do this without resorting to compiling and running the code to find the answer!*
- Pay particular attention in your answers to how many times you 'check' your answer, after the first two attempts an increasing mark penalty applies (we recommend compiling your programs and visually inspecting the output before you copy/paste into the quiz answer box)

When you've completed the quiz, use the rest of the lab to work on the remainder of this problem sheet.

Part 2: Functions
For this problem set, we would like you to undertake two tasks. The first of these is to write a simple set of functions to process numbers so you get into the swing of things. The second is a simple game simulation based on a 'fruit machine' that will get you practicing problem decomposition and choosing your own functions and functionality.

2.1 Number Processing Functions

Write functions that carry out the following operations:

- **int** multByTwo (**int**) —returns double the number passed to it.
- **int** isFirstBigger (**int**, **int**)—returns 1 if the first number is greater than the second, 0 otherwise.
- **float** addVat (**float**) —returns the parameter inflated by 20 percent.

For each set of functions we want you to write the functions and a main program that will show off your nicely constructed set of functions.

2.2 SCC.110 Fruit Machine

The SCC.110 fruit machine has 3 reels, each containing the following symbols:

1. a bell
2. an orange
3. a cherry
4. a horseshoe

Your goal is to successfully model the operation of the fruit machine, specifically:

1. Spin the reels (i.e. pick which symbol is to be shown on each reel)
2. Print out the reels (we suggest as text, e.g. `bell - orange - cherry`)
3. Detect whether the player has won (the player wins when two or more symbols match, jackpot is when all three symbols match)

You should think carefully about how to model your solution as a set of functions.

Hacker edition

The advanced task this week is to add a bit of reality to the game.

1. Instead of random reels, model the reels as a rotating drum of symbols that spin (these can be numbers, letters or ASCII symbols)
2. Display the reels as they spin by printing out the intermediate stages of the fruit machine (clearly this will be limited by the lack of graphics and terminal capabilities!)
3. Add a nudge function (i.e. the user can select a reel to nudge by 1 position)