Practical class 5

General comments

The objectives of this fifth lab session are:

(i) To understand how complex programs should be split into separate functions.
(ii) To practise writing simple functions with a range of parameters and return types.
(iii) To use functions appropriately in order to write a program with a menu system that can be used to determine if a number is prime or perfect.

Instructions

1. Write a program that, in addition to main(), contains three functions:

   a) `double square(double num)` which takes a double parameter and returns its square.

   b) `int isOdd(int num)` which takes an integer parameter and returns 0 if `num` is even and 1 if it is odd.

   c) `void printCircleProps(double radius)` which takes a radius and prints out the diameter, circumference and area of the corresponding circle.

Note that none of these functions needs to read input from the user (e.g. with `scanf`); they simply work with the argument that is passed. Similarly, only the last function needs to print anything; the first two instead return their result to the caller function with a `return` statement.

Carefully check all three functions by calling them from `main()`. An important part of good coding is extensive checking – try hard to break your functions by giving them hard/unusual input and fix them if they fail. For example, what happens if you pass a negative radius to the `printCircleProps` function?

2. Write a program that lets the user choose between three options:

   a) Test whether a number is prime.

   b) Test whether a number is perfect.

   c) Quit the program.

See https://en.wikipedia.org/wiki/Perfect_number if you don't know what a perfect number is. Write separate functions to determine if an integer is prime or perfect, and then call these as needed. The program must respond appropriately if the user input is invalid (e.g. if a negative number is entered).

3. Using your functions from part 2, write a program to print out all the prime numbers less than 1,000 and all the perfect numbers less than 10,000. (If you find any odd perfect numbers, please let me know.)

4. OPTIONAL EXTRA. Practise the use of functions and function calls by playing John Rowe's quiz at http://newton.ex.ac.uk/teaching/resources/jmr/quiz4.html.