## U08008 Modern Computer Technology – Example Assignment 1

In this assignment we will be testing your ability to solve a problem and write a program that implements your solution. Marks will be awarded for a correct answer that shows understanding of the python concepts covered. So for example, in a program where four identical actions were to be carried out, using a loop instead of just putting the same code four times would gain more marks.

**The Problem**

Dealing with a set of values that come in one at a time is a common computing task. The task here is to design, implement and test a program that will allow the user to enter a range of values terminated by a 0. Once the value 0 has been entered the program should display the number of even values entered.

**What you need to do**

You should first design the program informally. The design you submit should take the form of an explanation of why your program is structured the way it is and how the code works. It should show what data will need to be used by the program and detail the steps to use it.

You should next create a well structured python program that uses a lot of the concepts that we have learned up to this point and makes sensible use of comments.

You should carry out informal testing of your program. You should test with more enough sets of numbers to cover all the different sorts of situation that can occur. For example, three obvious situations are no even numbers, only even numbers and a spread of even numbers. To check thes three would need three tests each with a different set of test numbers

1. one test where there is no even numbers,
2. one test where there is only even numbers
3. one test where there are a few even numbers

The mathematical calculation is very simple but should still be checked. So is the program returning the correct number of even numbers for each of the cases. Work out the answer for yourself first then check the program produces the same answer.

**Submission**

You should submit the following:

1. Your informal design of your program. **[6 marks]**
2. A .py file containing your python code **[15 marks]**
3. Your informal testing **[4 marks]**

The work you submit should comprise of two files. A Word file (or equivalent that can be read from Word in the pooled rooms) and a .py file. The word file should contain the design, testing and the code copied and pasted in. The .py file should be a working python file containing your code.

This is an example assignment to show the format expected. It is a simpler assignment than the actual one set but the process and formats used are the same as expected.

## Answer

**Design**

The program will need to store two pieces of information, the next number (number) that has been entered and the amount of even numbers so far (numEvens).

As the number of values to be entered by the user is not known, the program will need a loop that will keep going until a zero is entered. In the loop the program will need to decide if the new value entered even or not and read another value

while number is not 0

update counter if number is even

input new number

In the check, if number divided by 2 has a remainder of 0 the number is even

if number % 2 == 0:

increase numEvens by 1

Once the loop has finished, the value of numEvens can be displayed on the screen.

Before the loop begins, both numEvens and number need to be given initial values.

As the loop compares against the value of number, this will need to be given a value before the loop starts. There are two possibilities for this. Either input a value into number before the loop starts or declare the variable with a value that isn't 0 so that the check on the loop will work and the loop will begin. As the first thing that happens in the loop is to check the value of number against the value of highest, the value in number at the start of the loop matters so the program will read a value into number before the loop starts.

As numEvens is a counter, it should start from 0

The program will do the following

input value into number

set numEvens to 0

while number is not 0

if number % 2 == 0:

increase numEvens by 1

input value into number

display the value of evens

**Code**

number = int(input(“enter the first number: “))

numEvens = 0

while number != 0:

if number % 2 == 0:

numEvens += 1

number = int(input(“enter the next number “))

print (numEvens, “ even numbers were entered”)

**Testing**

There will be the three cases discussed in the specification.

1. Where there is no even number
2. Where there are only even numbers
3. Where there is a mix

Data Expected Result Actual result

Test 1: 5, 7, 3, 13, 9, 3, 11, 0. numEvens should be 0 numEvens = 0

Test 2: 8, 2, 6, 4, 20, 6, 18, 8, 0 numEvens should be 8 numEvens = 8

Test 3: 2, 3, 8, 1, 11, 12, 8, 9, 0 numEvens should be 4 numEvens = 4