IDC101 (Introduction to Computation) : Lab session 4

General Instruction :
For each problem given here, first decide your algorithm. That is, first decide how you will solve the problem on paper. Only after this, you should start writing your program.

Here's the checklist. DO NOT proceed without following these steps.
a) Read the problem carefully. Decide the inputs required.
b) Decide your step-by-step algorithm. That is, decide how you will solve your problem step-by-step.
c) Now, write your program following your algorithm.

---

1. Write a program which will add the integers from 0 to 10. Check if your answer is correct.

Brief sample algorithm :

No input is needed for this program.

Initialise `sum = 0`.
Loop variable is `n`. Initialise `n=1`.

Use `while` loop to compute the result..

Write the output in desired form.

2. Generalise the program given in problem 1 to compute the sum of integers up to any specified value *N*. In this case, the value of *N* must be read as input to the program. Solution is given on the course website in the program `loop1.py`.

3. Write a program to find the sum of squares of all the odd integers less than 20. Modify this program to find the sum of squares of all the even integers less than 20.

4. Write a program to compute the result of following summation :

$$S = \sum_{n=1}^{10} \sin^2(\pi/n).$$

5. Create a list of ten real numbers. Write a program that will give the maximum of these ten numbers as output. The program should correctly handle positive and negative numbers. The output should also include the input numbers, apart from the result.

Brief sample algorithm :
Create a list of 10 real numbers in the variable name `nums`.

Let `nmax` be the variable that will hold the largest number.
To begin, assume `nums[0]` to be the largest number.
Then, set `nmax = nums[0]`.

Use `while` loop to compare each element of `nums` to present value of `nmax`.
If present value is greater, `nmax` will be reset to the present value.
If not, continue to the next member of the loop.

When the loop finishes, write the last value of `nmax`.

6. Copy the program you wrote for problem 5 in to a different file. Then, modify this program to give the minimum of ten numbers as output.

7. Write a program to estimate the following summation to three digit accuracy;

$$S = \sum_{n=0}^{\infty} \frac{1}{2^n}$$

A sample program `loop2.py` to solve this problem is given on course website.

8. Write a program to compute the value of $\exp(1)$ correct to two decimal places. Figure out how to stop your loop precisely at two digit accuracy.

9. Write a program to compute the value of $\exp(-x)$ correct to two decimal places. Note that you have to take x as the input. Take x to be small number less than 1. For example, take $x=0.5$.

10. Use the following Gregory-Madhava series to compute π correct to 3 decimal places.

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \ldots.$$