# Department of Computing

# Fundamental of Computer Programming

# Class: SE-7B

# Lab 8: Loops

# Date: 30th November, 2016

# Time: 2pm-5pm

# Instructor: Muhammad Muddasir Malik

# Lab 8: Loops

# Introduction

A loop statement allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages: C programming language provides the following types of loop to handle looping requirements.

**Tools/Software Requirement**

Python IDLE

**Description**  
  
WHILE - WHILE loops are very simple. The basic structure is   
  
while ( condition ) { Code to execute while the condition is true } The true represents a boolean expression which could be x == 1 or while ( x != 7 ) (x does not equal 7). It can be any combination of boolean statements that are legal. Even, (while x ==5 || v == 7) which says execute the code while x equals five or while v equals 7. Notice that a while loop is like a stripped-down version of a for loop-- it has no initialization or update section. However, an empty condition is not legal for a while loop as it is with a for loop

**Tasks:**

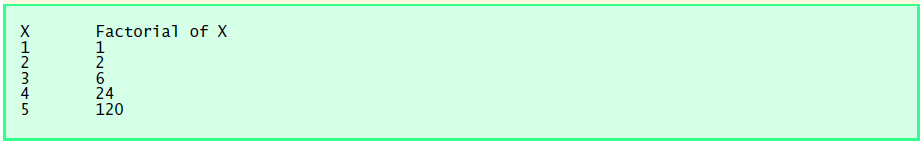
1. Write a program that sums a sequence of integers. Assume that the first integer read with the input function specifies the number of values remaining to be entered. Your program should read only one value with each input function call. A typical input sequence might be the following where the 5 indicates that the subsequent five values are to be summed.

5 100 200 300 400 500

1. Write a program that finds the smallest of several integers. Assume that the first value read specifies the number of values remaining.
2. The factorial function is used frequently in probability problems. The factorial of a positive integer n (written n! and pronounced “n factorial”) is equal to the product of the positive integers from 1 to n.

Write a function factorial that accepts an integer as parameter and returns its factorial.

Using the factorial function, write a program that evaluates the factorials of the integers from 1 to 5. Print the results in tabular format as following.



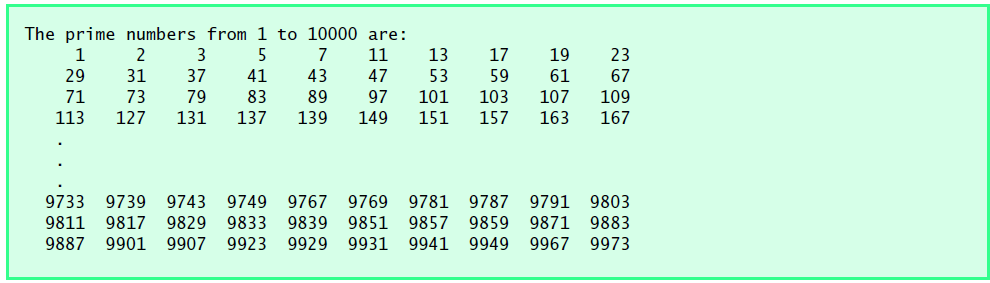
1. Write a program that plays an incredibly stupid number-guessing game. The user will try to guess the secret number until they get it right. That means it will keep looping as long as the guess is different from the secret number. You must store the secret number in a variable, and use that variable throughout. The secret number itself must not appear in the program at all, except in the one line where you store it into a variable. Sample output is as following:

|  |
| --- |
| I have chosen a number between 1 and 10. Try to guess it.  2 Your guess: 5  3 That is incorrect. Guess again.  4 Your guess: 4  5 That is incorrect. Guess again.  6 Your guess: 8  7 That is incorrect. Guess again.  8 Your guess: 6  9 That's right! You guessed it. |

1. An integer is said to be prime if it is divisible only by 1 and itself. For example, 2, 3, 5 and 7 are prime, but 4, 6, 8 and 9 are not.

a) Write a function that determines if a number is prime.

b) Use this function in a program that determines and prints all the prime numbers between 1 and 10,000.



1. Write a program that reads a number (Between 1 and 30). Your program should print a line containing that number of adjacent asterisks. For example, if your program reads the number seven, it should print \*\*\*\*\*\*\*.