Assignment 1: Programming in Python CST 362

Name: **Alby Thekkedan Roll No 8 Class CS6B**

KTU-ID: **MDL20CS129**

Date of submission: 27-Feb-2023

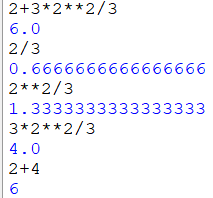
**Learning outcome: Python basics, operators, modules/packages, control statements**

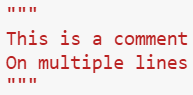
1. Python is developed by **Guido van Rossum**
2. How Python got its name - **The name, Python comes from an old BBC television comedy sketch series called Monty Python's Flying Circus**
3. Write the output of the statement 2\*\*3\*\*2. **Output - 512**
4. What is the difference between / and // operators. Write one example

**‘ / ’ is Normal division the output will be a real value or floating value. eg: 15/4=3.75**

**‘ // ’ is Integer division or Floor division divides and returns integer value eg: 15//4=3**

1. How the expression 2+3\*2\*\*2/3 is evaluated show the order of evaluation and final output

**Ans : 6** 

1. ****How single line and multiline comments are added in Python script

**To make single-line comments in Python, add # to the beginning of each line. #a=7**

**To make multi-line comments in Python, Use docstrings**

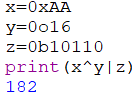
1. Write the logical operators

They are **and, or, not**

The **and**operator checks whether **two conditions are both True simultaneously.** It returns True if both conditions are True. And it returns False if either the condition is False.

The **or** operator returns True when **either or both individual conditions are True.** The or operator returns**False**only when **both conditions are False**.

The **not**operator **If the condition is True, the not operator returns False and vice versa.**

****

1. x=0xAA, y=0o16, z=0b10110 , find x^y|z

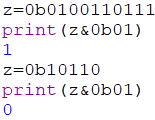
**x is (170)10, y is (14)10, z is (22)10**

**Ans 182**

1. What are bit wise operators

|  |  |  |  |
| --- | --- | --- | --- |
| **OPERATOR** | **NAME** | **DESCRIPTION** | **SYNTAX** |
| & | Bitwise AND | Result bit 1, if both operand bits are 1; otherwise results bit 0. | x & y |
| | | Bitwise OR | Result bit 1, if any of the operand bit is 1; otherwise results bit 0. | x | y |
| ~ | Bitwise NOT | Inverts individual bits | ~x |
| ^ | Bitwise XOR | Results bit 1, if any of the operand bit is 1 but not both, otherwise results bit 0. | x ^ y |
| >> | Bitwise right shift | The left operand’s value is moved toward right by the number of bits specified by the right operand. | x>> |
| << | Bitwise left shift | The left operand’s value is moved toward left by the number of bits specified by the right operand. | x<< |

1. a. How to get the last bit of a number. Write the bitwise operation, e/g: x=2 o/p:0, x=3 o/p:1

b. x=12, what is the output of x<<2 justify your answer

 **BITWISE AND operation with binary 01**

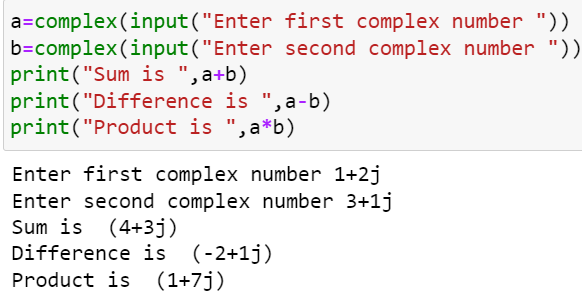
**12 is 1100, left shift by 2 position on left most bit**

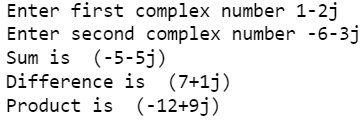
**gives 110000, which is (48)10**

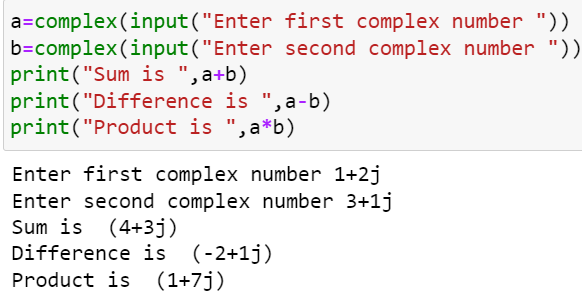
1. What are Boolean data types

**The Boolean value can be of two types only i.e. either True or False**.**It is used to represent the truth values of the expressions. For example, 1==1 is True whereas 2<1 is False.**

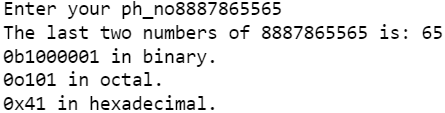
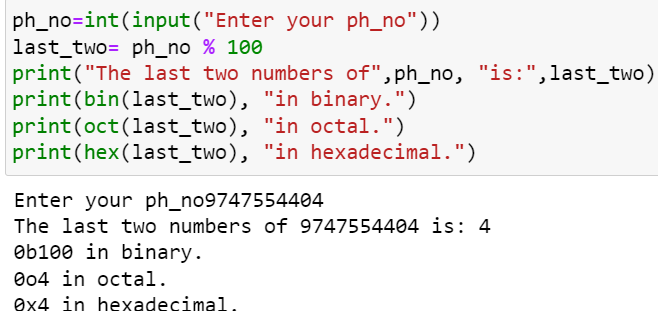
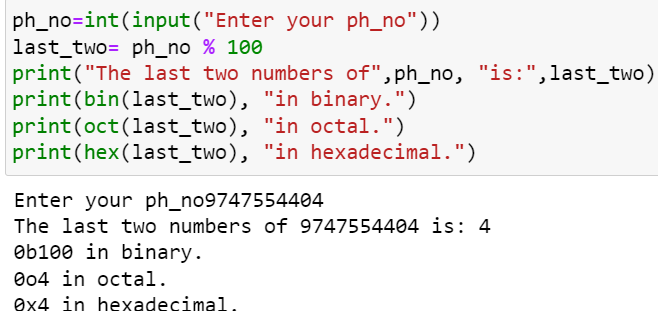
**Basic Scripting**

1. Write a Python script which will read two complex numbers and find their sum, difference and product.

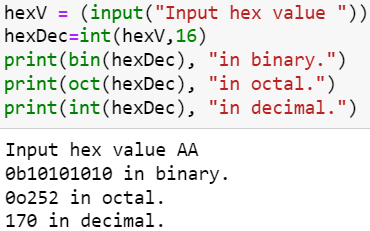
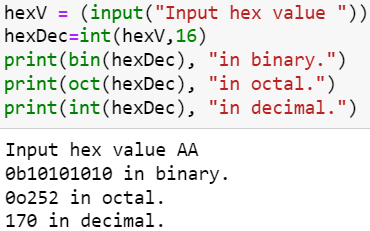
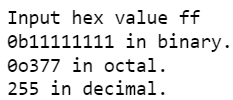
****



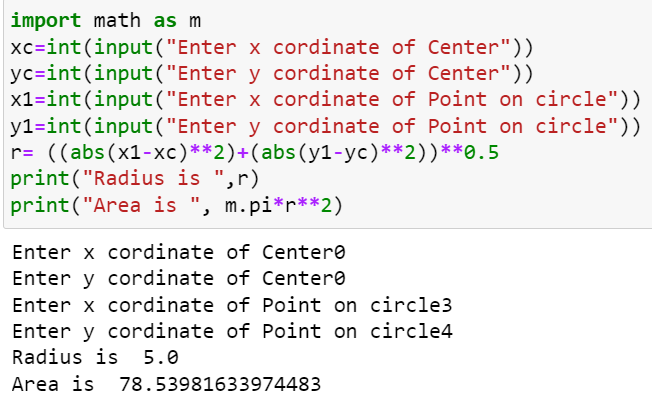
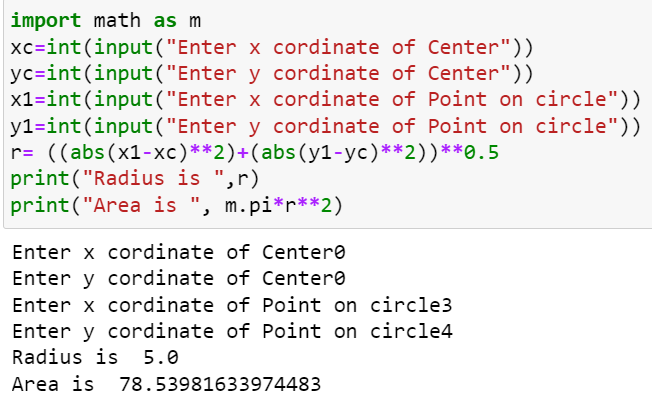
1. Read your phone number and print the last two digit in binary, octal and hex.

****

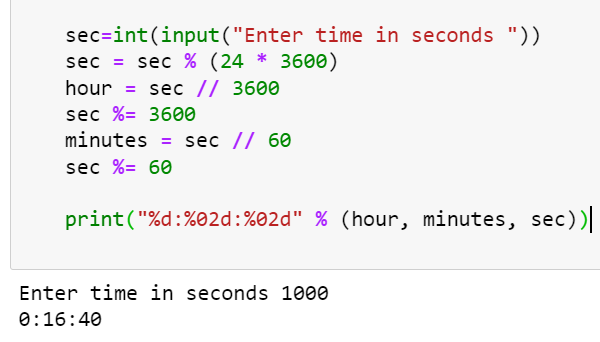
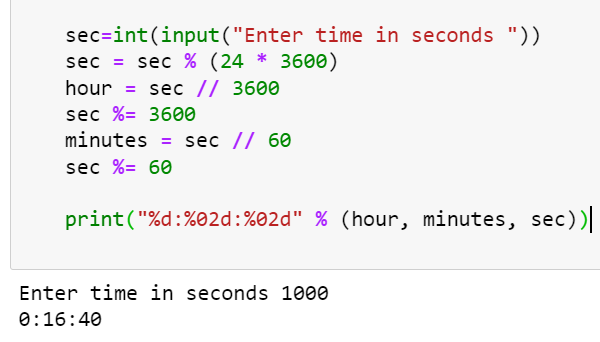
1. Read a hexa decimal number and print the dec, bin and octal equivalent.



1. Given the center (xc,yc) and a point on the circle(x1,y1). Find the area

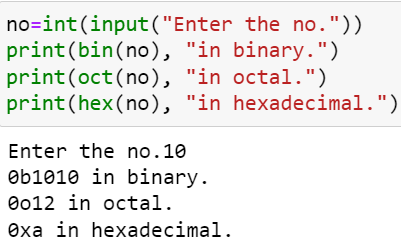
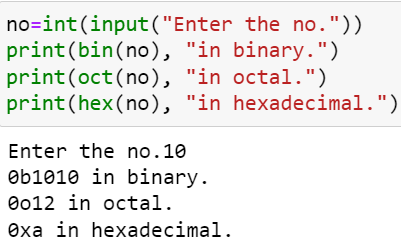
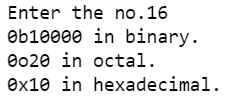


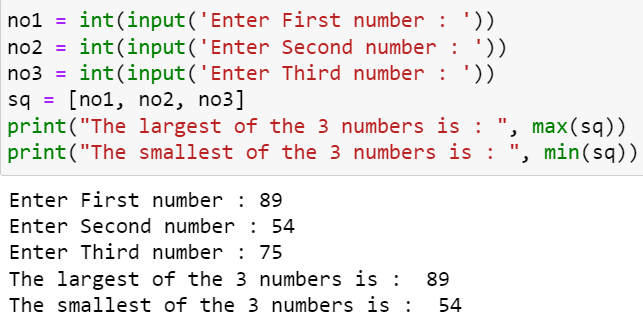
1. Write a Python program to read time in seconds and Print in HH:MM:SS format. i/p in seconds :1000 o/p:00:16:40

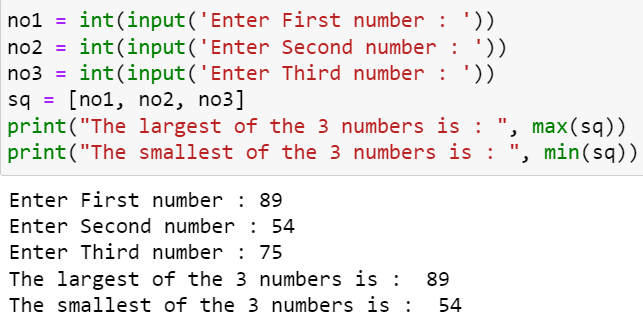


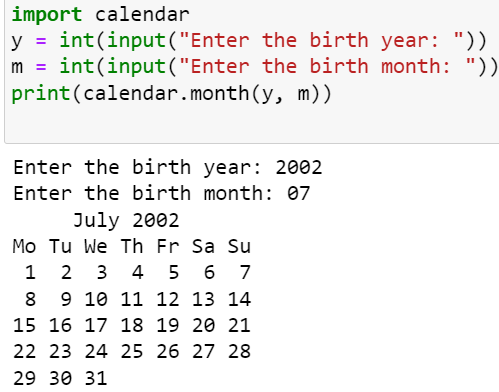
**Use built in functions**

1. Read a number and Print the corresponding binary, oct, hex

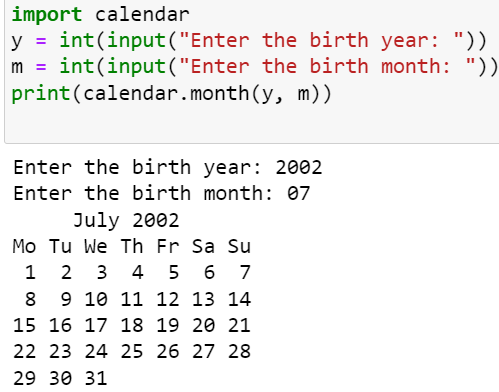


1. Read 3 numbers and find the largest and smallest

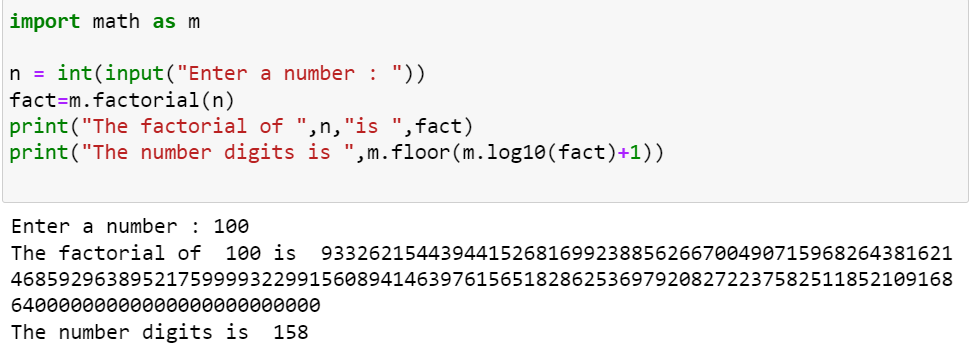
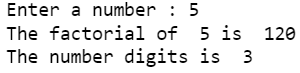


**Use modules/packages**

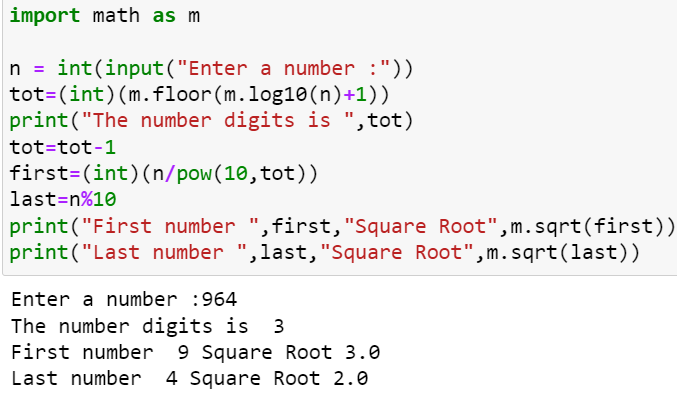
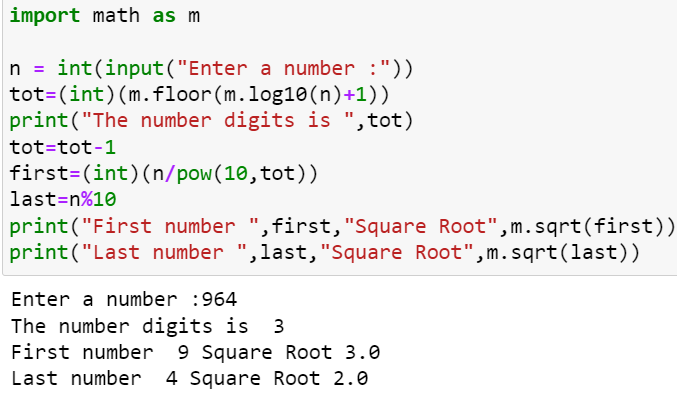
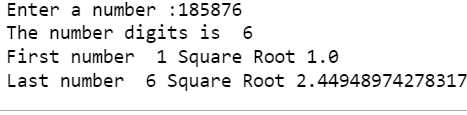
1. Print the month calendar depending on your DOB



1. Find the number of digits in the factorial of a given number

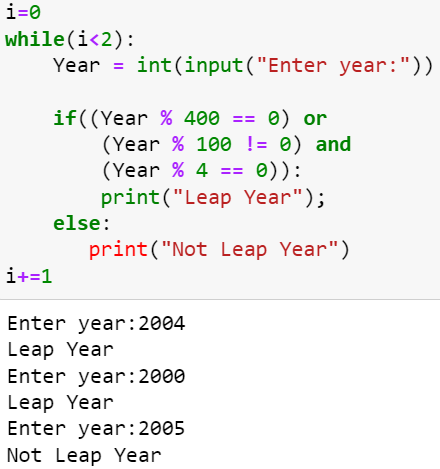
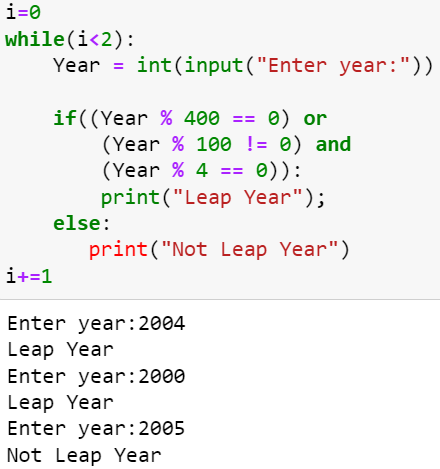


1. Find the sqrt of first and last digit of a number

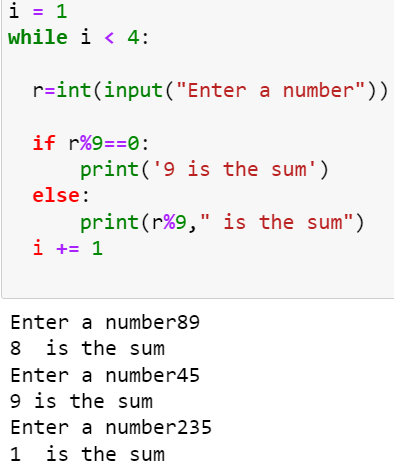
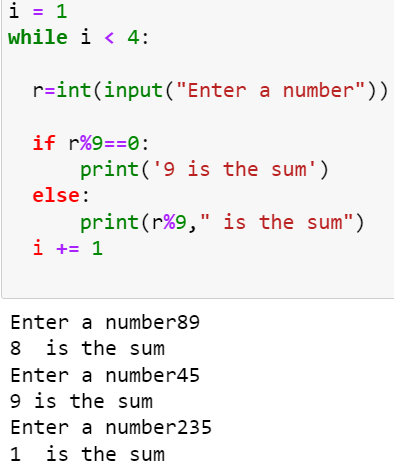


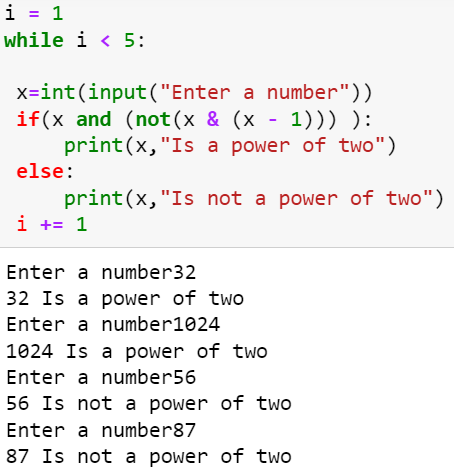
**Using if**

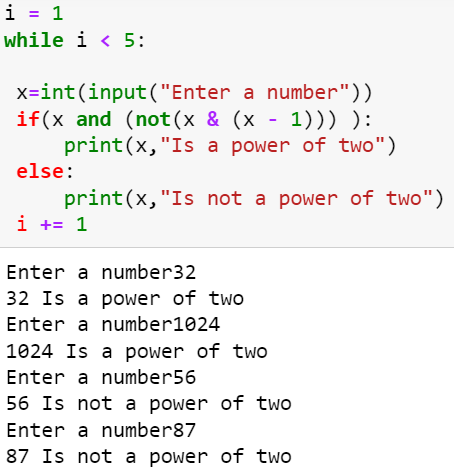
1. Check whether the given year is leap year or not.



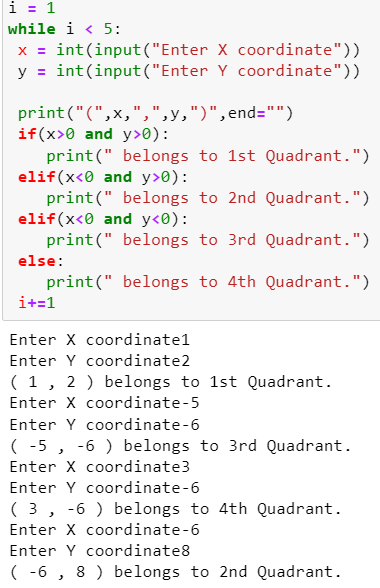
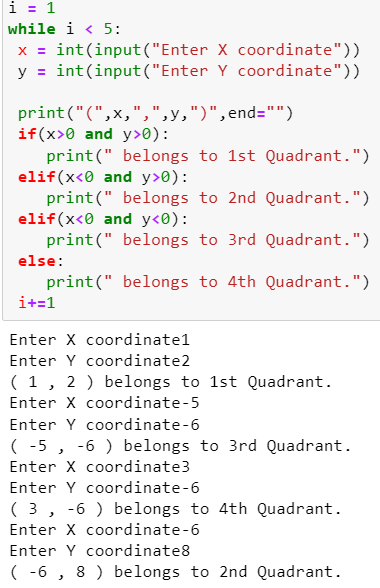
1. Write a python program to read a number and recursively add the digits in it



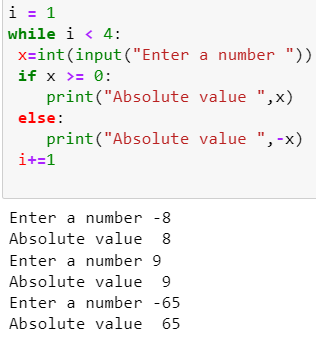
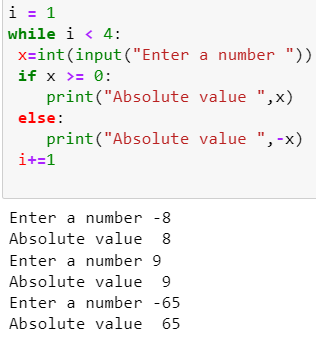
1. Check whether the given number is power of 2



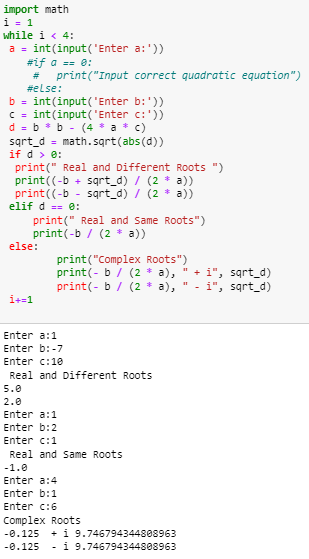
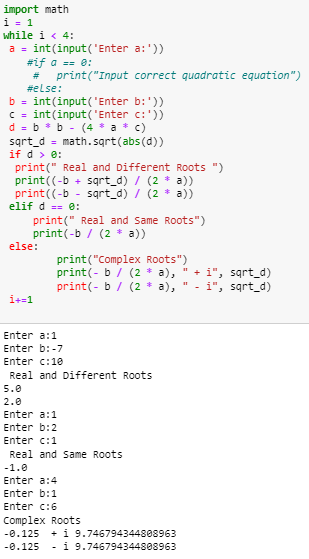
1. Write a program to check the quadrant of a given point(x,y)**( University question)**

****

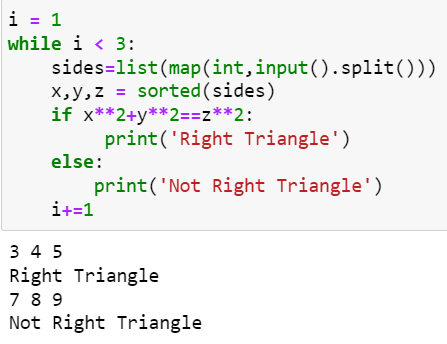
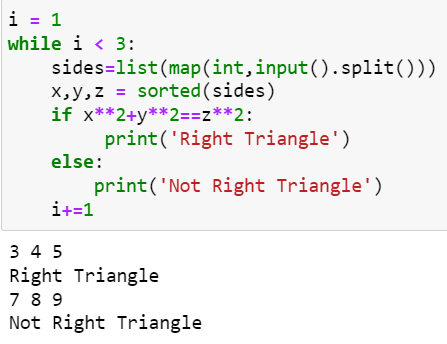
1. Write a program to get the absolute value of a number without using the abs() function**.(university question)**



1. Find the roots of a quadratic equation

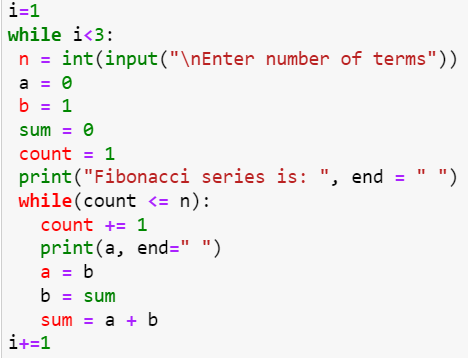
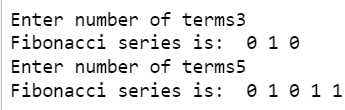


1. Write a program that accepts the length of three sides of a triangle as input and determine whether or not the triangle is a right triangle**.( university question)**

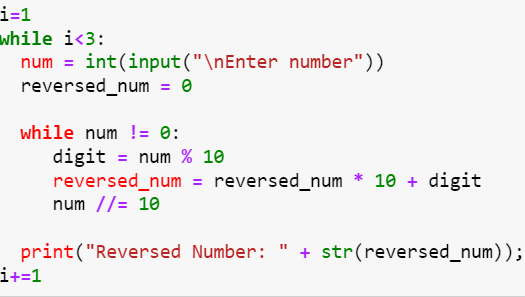
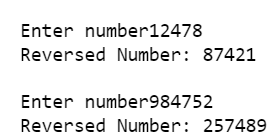
****

**Using while/for**

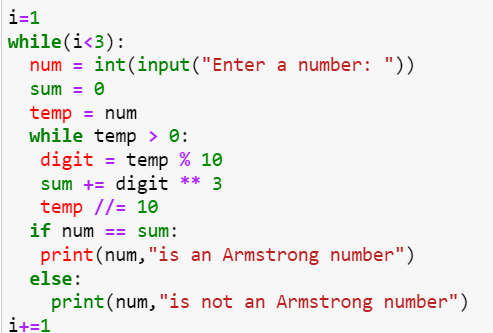
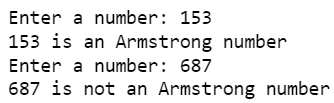
1. Generate the Fibonacci series 0 1 1 2 3 5 8…..n ( use while..read n)



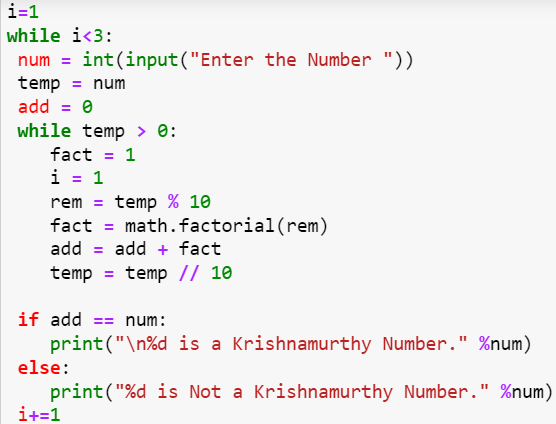
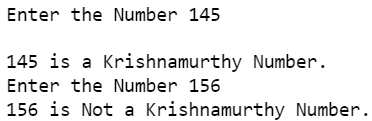
1. Reverse a Number ( i/p: 123 o/P 321)



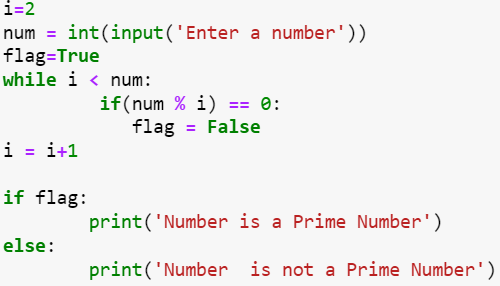
1. Check whether the given 3 digit number is **Armstrong Number**. Eg:153 ( 1+125+27=153) sum of cubes of digits=number.



1. Check whether the given number is a **Krishnamurthy number**

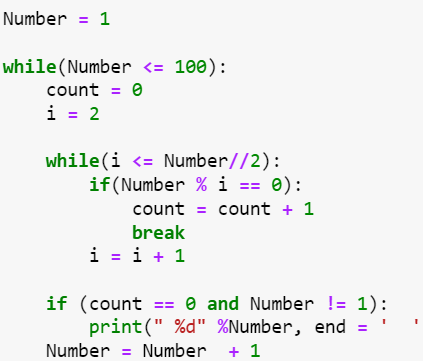


1. Check whether the given number is **Prime or not**



**Nested Loops(for/while**)

1. Print all prime numbers less than 100.



1. Print the binary equivalent of each digit of the given number.

