## ASSIGNMENT - I

## NAME : PARISHKAR SINGH

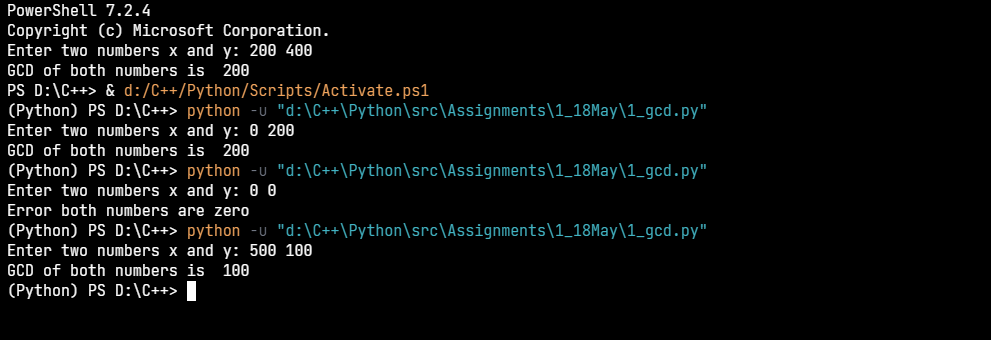
## USN : 2GI20CS081

## DATE - 25- MAY-2022

## DIV: 4-B

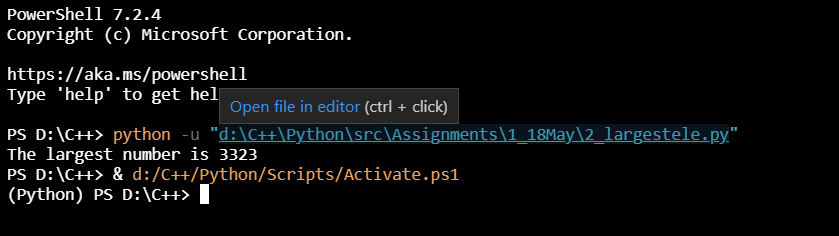
## *Q1. Write a program to compute GCD of t wo numbers. It should check for both non zero, and if a<b should exchange and use. If both zero, report error and exit. Write a function GCD(a,b) that returns the GCD of a and b as result.*

*def* GCD(x, y):  
 *if* x == 0 *and* y == 0:  
 *return False  
 # raise Exception("Error, both numbers are zero")  
 if* x == 0:  
 *return* y  
 *if* y == 0:  
 *return* x  
 *if* x == y:  
 *return* y  
 *if* x > y:  
 *return* GCD(x-y, y)  
 *return* GCD(x, y-x)  
  
  *if* \_\_name\_\_ == '\_\_main\_\_':  
 a, b = map(int, input("Enter two numbers x and y: ").split())  
 *if* GCD(a, b):  
 print("GCD of both numbers is ", GCD(a, b))  
 *else*:  
 print("Error both numbers are zero")



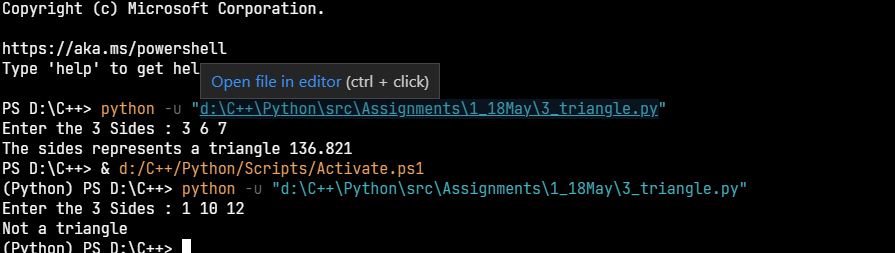
## *Q2. Write a program to find the largest element in the Array. Do it by writing function do this*

*def* largest(list):  
 x = 0  
 *for* i *in* list:  
 *if* i > x:  
 x = i  
 *return* x  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 list = [12, 213, 234, 3, 45, 756, 67, 3323, 4]  
 print("The largest number is",largest(list))



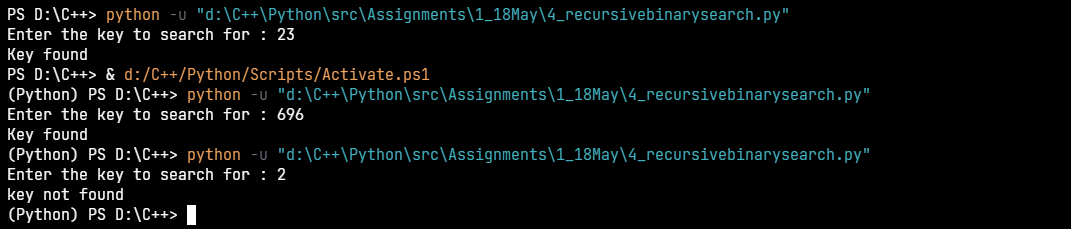
## *Q3. Write a program read three values from keyboard as 3 sides of a triangle. Check if they form triangle. If yes compute area else report as error input.*

*import* math  
  
*def* checktriangle(a, b, c):  
 *if* a >= b+c *or* b >= a+c *or* c >= a+b:  
 *return False  
 return True  
  
def* computerarea(a, b, c):  
 s = a+b+c  
 *return* math.sqrt(s\*(s-a)\*(s-b)\*(s-c))  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 a, b, c = map(int, input("Enter the 3 Sides : ").split())  
 *if* checktriangle(a, b, c):  
 print("The sides represents a triangle %4.3f" % computerarea(a, b, c))  
 *else*:  
 print("Not a triangle ")



## *Q4. Write a program to implement recursive binary search to search for key in list!.*

*def* binarysearch(arr, start, end, key):  
 *if* start < end:  
 mid = int((start+end)/2)  
 *if* arr[mid] == key:  
 *return True  
 elif* arr[mid] > key:  
 *return* binarysearch(arr, start, mid, key)  
 *elif* arr[mid] < key:  
 *return* binarysearch(arr, mid+1, end, key)  
 *return False  
  
  
if* \_\_name\_\_ == '\_\_main\_\_':  
 arr = [23, 50, 183, 219, 233, 345, 657, 696]  
 key = int(input("Enter the key to search for : "))  
 *if* binarysearch(arr, 0, len(arr), key):  
 print("Key found")  
 *else*:  
 print("key not found ")



## Q5. Write a python program to print the following

## \*

## \* \*

## \* \* \*

## \* \* \* \*

*def* printpattern(a):  
 *for* i *in* range(0, a+1):  
 *for* j *in* range(0, i):  
 print("\*", end=" ")  
 print()  
  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 a = int(input("Enter the number of the rows: "))  
 printpattern(a)

