



JECRC™  
UNIVERSITY  
BUILD YOUR WORLD

JAIPUR ENGINEERING COLLEGE AND RESEARCH CENTRE, JAIPUR

DEPARTMENT OF COMPUTER APPLICATIONS

**LAB FILE**

**NAME: VALLABH NARULA**

**REGISTRATION NO.: 23BCAN0087**

**DEGREE: BCA (GENERAL)**

**SEMESTER: 2nd**

**SECTION: A**

**SUBJECT CODE: BCA301A**

**SUBJECT: PYTHON**

**FACULTY: PROF. AMAR VERMA**

**(ASSISTANT PROFESSOR-JECRC UNIVERSITY)**

**ACADEMIC YEAR: 2023-2024**

# Index

SI No.	Name of the Experiment	Page No.	Date of Experiment	Date of Submission	Remarks
1	GCD of 2 numbers		5/2/24	19/2/24	Done
2	Sqrt of a number		5/2/24	19/2/24	Done
3	Find the exponential		12/2/24	19/2/24	Done
4	Find largest in list		12/2/24	19/2/24	Done
5	Find using linearSach.		12/2/24	19/2/24	Done
6	Binary Search		19/2/24		Done
7	Prime number		26/2/24		Done
8	Even odd		26/2/24		Done
9	Area of Triangle		4/3/24		Done
10	Distance B/W Two Point		4/3/24		Done
11	Bill amount of item		4/3/24		Done
12	Date and time		1/4/24		Done
13	Working with Tuple		1/4/24		Done
14	Working with Dict		1/4/24		Done
15	math Built-in Function				
16	use of userDefinedModel				
17	operation on Dictionaries				
18	operation on Tuples				
19	Number Data Types				
20	Arithmetic operators in numbers				

Subject		Class	
Semester		Roll No.	

# INDEX

Serial No	Date	Title	Page No	Teacher's Signature/Remarks
21		Module To find Fibonacci No's		
22		Create, Concatenate / print String		
23		Refine module / import func		
24		Print Prime No's		
25	6/5/24	factorial using recursion		
26	6/5/24	Math Built-in Function		
27	6/5/24	String Built-in Function		
28	6/5/24	Convert Temp To Fahrenheit		10/15
29	6/5/24	Print unique words in Text file		
30	6/5/24	Int To Float value		
31	6/5/24	class To Implement flow		
32	6/5/24	Program using for loop		
33	6/5/24	Program using while loop		
34	6/5/24	User Defined Expression		
35	6/5/24	Simple / Multiple Inheritance		

Experiment-1

Q1 Find out the GCD of Two numbers using Python

→ ~~Import math~~

```
a = int(input("Enter first value"))
b = int(input("Enter second value"))
get = (math, gcd(a,b))
```

~~print ("GCD is {get}")~~

~~or~~

~~import math~~

~~a = 10
b = 20~~

~~get = (math, gcd(a,b))~~

~~print ("GCD is " + get)~~

\* ~~gcd → The math.gcd() method is used to find the greatest common factor or highest common factor.~~

output

Enter First Value : 10

Enter Second Value : 20

GCD is 10

output :

GCD is 10

## Experiment-2

objective → WAP To find Sqrt of a number

Code :

(i) Without using pow()

```
a = int (input ("Enter a number:"))  
result = a ** 2
```

```
print ("Sqrt of day is : " ; result)
```

Using pow()

```
def Sqrt (number)  
if number <= 0: // if negative number  
    print ("Enter a Positive no :")  
return pow (number, 0.5) // Sqrt formula
```

```
number = int (input ("Enter a numbers : "))  
result = Sqr.root (number)  
print ("The Sqrt of {} number is {} ".format (number, result))
```



✓ 100

Output 1 :

Enter a number : 5

Square root of 5 is 25

Output 2 :

Enter a no : 5

The Square root of 5 is 25

### Experiment - 3

Q Find the exponential

a = int(input("Enter the number:"))

b = int(input("Enter the exponential:"))

$$\text{exp} = a * b$$

~~print(exp)~~

OR

Import math

a = int(input("Enter the number:"))

~~b = int(input("Enter the exponential:"))~~

$$\text{exp} = \text{math.pow}(a,b)$$

~~print(exp)~~

Exponentials :- Containing or involving one or more numbers or quantities raised to an exponent

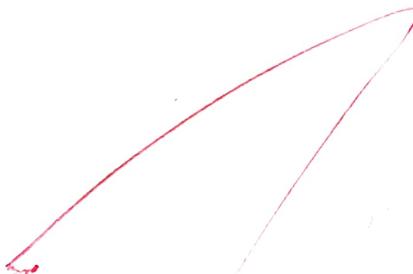
For ex:  $2^4, 3^3, 8^2$

Output :-

Enter the number : 5

Enter the Exponential : 2

25



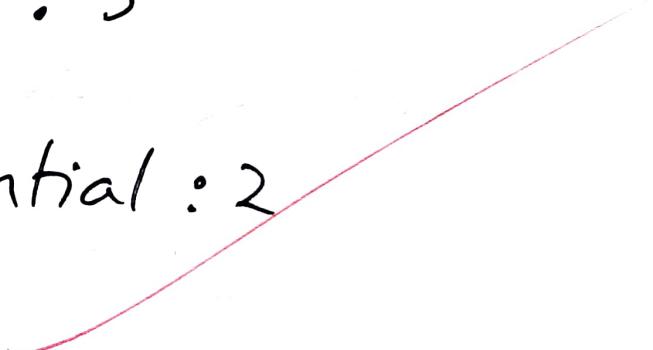
Output :-

Enter the number : 5

~~Exit~~

Enter the exponential : 2

25



## Experiment - 4

WAP in Python To find the largest in a list

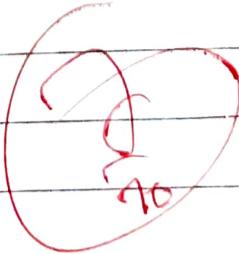
~~a = [10, 20, 30, 40, 50, 80]~~

~~max = max(a)~~

~~print ("Largest number is : ", max)~~

\* A list is a data structure in Python that is a mutable or changeable, ordered sequence of elements.

\* ~~max() → max() function is a built-in tool that returns the largest term in an iterable or the largest of two or more arguments.~~



X 70 ✓

output

Largest no. is : 80

## Experiment-5

Objective : WAP To find a value using linear search in Python.

Code :

```
def ls (lst, n, x):
    for i in range (0, n):
        if (lst [i] == x):
            return i
    return -1.
```

~~lst = [1, 2, 3, 4, 4, 5, 6, 7, 8]~~

~~x = 4~~

~~n = len (lst)~~

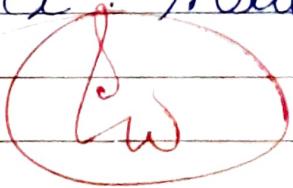
~~result = ls (lst ,n, x)~~

~~if (result == -1):~~

~~print ("Element not found")~~

~~else :~~

~~print("element found at index : ", result)~~



M.: 1877427231

output :-

Element found at Index : 3

## Experiment-6

objective : To find a value using Binary Search

```

def binary_search(arr, a, low, high):
    while low <= high:
        mid = low + high // 2
        if arr[mid] == a:
            return mid
        else:
            high = mid - 1
    return -1

```

```

arr = [9, 7, 5, 3, 1]
a = int(input("Enter a number"))
print("Element is to be found is", a)
index = binary_search(arr, a, 0, len(arr)-1)
if index != -1:
    print("The index of the element is" + str(index))
else:
    print("Element not found")

```

88M

6  
6

Output

The array given is [9, 7, 5, 3, 1]

Enter a number : 5

Element to be found is 5

The index of the element is 2

## Experiment - 7

Objective : WAP to print Prime number

```
def is_prime(number):  
    if number <= 1:  
        return False
```

```
for i in range(2, int(number ** 0.5) + 1):  
    if number % i == 0:  
        return False  
return True
```

# Test the function

```
num = 17  
if is_prime(num):  
    print(f"\{num} is a Prime number")  
else:  
    print(f"\{num} is not a Prime number")
```

(6)

Output :

17 is a Prime number

## Experiment-8

Objective: WAP to print even odd

Code:

~~num = Int [Input("Enter a number:")]~~

~~If (num % 2 == 0):~~

~~print ("Prime number")~~

~~else:~~

~~print ("Not Prime number")~~

Even number = Number is Divisible by 2

Odd number = Number is not Divisible by 2

In this Program. The user is Taking Input from the user and then give the result as per the condition.

QW

Output:

Enter a number : 3

not prime number

## Experiment - 9

Objective : WAP to calculate Area of triangle using Heron's formula.

Code :

~~Import math~~

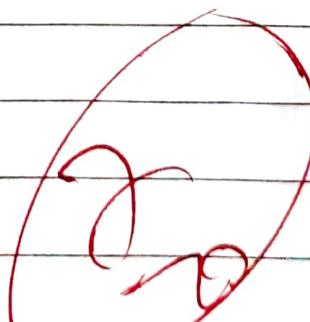
~~a = Int(Input("Enter side 1 : "))~~

~~b = Int(Input("Enter Side 2 : "))~~

~~c = Int(Input("Enter Side 3 : "))~~

~~S = (a + b + c) / 2~~

~~area = math.sqrt(S \* (S - a) \* (S - b) \* (S - c))~~



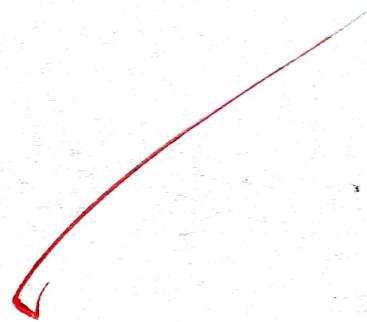
Output

$$\text{Side 1} \geq 5$$

$$\text{Side 2} \geq 5$$

$$\text{Side 3} \geq 5$$

$$\Rightarrow 10.82$$



## Assignment - 10

Objective : WAP to calculate distance  
b/w Two Points

Code :

~~Import Math~~

~~$x_1 = \text{float}(\text{input}("Enter x_1 = "))$~~   
 ~~$x_2 = \text{float}(\text{input}("Enter x_2 = "))$~~

~~$y_1 = \text{float}(\text{input}("Enter y_1 = "))$~~   
 ~~$y_2 = \text{float}(\text{input}("Enter y_2 = "))$~~

$\text{area} = \text{math.sqrt}((x_2 - x_1)^2 + (y_2 - y_1)^2)$   
print (area)

(2)  
2n

Output:

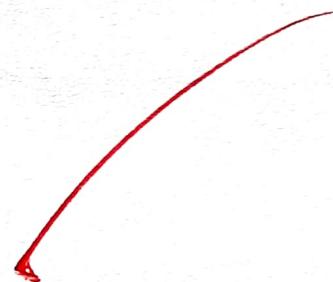
$$x_1 = 3$$

$$x_2 = 4$$

$$y_1 = 3$$

$$y_2 = 3$$

100



## Experiment-11

Objective: WAP to calculate all Bill Amount of an item given its quantity sold, value discounts & tax.

quantity = float(input("Enter quantity of item sold :"))

value = float(input("Enter value of item"))

discount = float(input("Enter discount percentage :"))

Tax = float(input("Enter the Tax :"))

# Calculate Bill

amount = quantity \* value

discount\_amount = amount \* (discount / 100)

subtotal = amount - discount\_amount

total\_amount = Subtotal + Tax amount

# Display the Bill

Print(Bill)

print("Quantity Sold:", quantity)

Output:

Bill

Quantity Sold	80
Price Per Item	100

Amount	8000
Discount	800
Discount Total	17200

Tax 1008

Total Amount 8208