

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PPLAB

Name K.SREE INDIRA SIVANI Roll No. 1602-21-733-052 Page No. 39

PRELAB QUESTIONS - 3

1) What are recursive functions in python? Give example:

A recursive function is defined as a function that calls itself to solve a smaller version of its task until a final call is made which does not require a call itself.

Recursive function has a base case and recursive case.

Ex: def fact(n):

```
if (n==0 or n==1):  
    return 1  
else:  
    return n*fact(n-1).
```

2) What are the rules for writing recursive function?

A: 1) Specify the base case which will stop the function from making a call 'itself'

2) Check to see whether the current value being processed matches with the value of the base case. If yes, process and return the value.

3) Divide the problem into a smaller or simpler sub-problem.

4) Call the function on the sub-problem.

5) Combine the results of the sub-problems.

6) Return the result of the entire problem.

3) Differentiate between iteration and recursive function.

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PP LAB

Name K. Sree Indira Savani Roll No. 1602-21-733-052 Page No. 40

- * Iteration: Iterative statements are decision control statements that are used to repeat the execution of list of statements.
- * Recursion: It is a function which calls itself to solve a smaller version of the main task and repeat it over every recursive call until the base case becomes false.
- * What are modules in python? Name few modules and examples of it:
Modules allows you to reuse one or more functions in your programs.
It is a .py file that has the definitions of all functions and variables that we would like to use in other program.
Eg: 1) sys module: path, argv, exit.
2) math module: pi, sqrt.

PRELAB PROGRAMS: 3

- 1) WAP to calculate the sum of 'n' natural numbers using recursive function:

~~def recu_natural(n):
 sum = 0
 for i in range(1, n+1):
 sum = sum + i
 return sum~~

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PP LAB

Name K. Sree Indira Sivani Roll No. 1602-21-733-052 Page No. 41

* def natural(n):
 sum=0
 if n<=1:
 return n
 else:
 return n+natural(n-1)

m=int(input("Enter a number:"))
Enatural(m).
print("Sum = ", r)

Output:

Enter a number: 16
Sum = 136

2) WAP to find the sum of digits of a number using recursive function:

* def s(n):
 if (n==0):
 return 0
 else:
 return (n%10 + s(n//10))

Output:

Enter a no: 1635
Sum of digits = 10

3) WAP to find the no. of digits in a given no. using recursive function:

* def digits(n):
 ans if (n!=0):
 count++

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PP LAB

Name K. Sri Indira Sirani Roll No. 1602-21-733-D52 Page No. 42

ddigits(n//10) return count. m = int(input("Enter a no:")) x = digits(m) print("No. of digits =", x)	<u>Output:</u> Enter a no: 173 No. of digits = 3
--	--

LAB PROGRAMS :-

1) WAP

* AIM: To illustrate the use of recursive functions and modules in day today programs.

1) WAP to calculate the factorial of a given no using recursive functions:

def factorial(n): if n==0 or n==1: return 1 else: return (n * factorial(n-1)) m = int(input("Enter a number:")) r = factorial(m) print("Factorial of", m, "is", r)	<u>Output:</u> Enter a number 5 Factorial of 5 is 120
---	---

2) WAP to print Fibonacci series using recursive function:

* def fib(n):
 if n<=1:
 return n
 else:
 return fib(n-1) + fib(n-2)
m=int(input("Enter a number"))
print('Fibonacci series')
for i in range(m):
 print(fib(i))

Output:

Enter the number 5

Fibonacci Series .

0

1

1

2

3

3) WAP to find the power of the given number:

* def power(x,n):
 if n==0:
 return 1
 elif n==1:
 return x
 else:
 return x*power(x,n-1)

Output:

Enter the base number 5

Enter the exponent 6

5 power 6 is 15625 .

m=int(input("Enter the base number"))

y=int(input("Enter the exponent"))

r=power(m,y).

print(m,"power",y,"is",r).

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PP LAB

Name K. Sri Indira Sivani Roll No. 1602-21-733-052 Page No. 44

- 4) WAP to calculate nCr ; using recursive function to calculate the factorial:

* import fact

```
n=int(input("Enter the number"))
r=int(input("Enter the number"))
x=(fact*factorial(n))/(fact*factorial(n-r)*fact*factorial(r))
print("Value=",x)
```

Output:

Enter the number 6

Enter the number 2

Value=15.0

- 5) WAP to check the type of \triangle (Equilateral, Isosceles, Scalene) using functions and by entering the vertices.

import math

```
def distance(a,b,c,d):
```

```
    return math.sqrt((a-c)**2 + (b-d)**2).
```

```
def check(x,y,z):
```

if $x==y$ and $y==z$:

```
    print("Equilateral triangle")
```

elif $x==y$ (or) $y==z$ (or) $z==x$:

```
    print("Isosceles triangle")
```

else:

```
    print("Scalene triangle")
```

```
print("Enter the first vertex")
x = int(input())
y = int(input())
print("Enter the second vertex")
u = int(input())
v = int(input())
print("Enter the third vertex")
s = int(input())
t = int(input())
l = distance(x,y,u,v)
m = distance(u,v,s,t)
n = distance(s,t,x,y)
check(l,m,n)
```

Output:

Enter the first vertex:

8

6

Enter the second vertex:

3

2

Enter the third vertex:

5

7

Scalene Triangle.

6) Write a program to implement Pascal's triangle

```
def pascal(n):
    for i in range(n):
        for j in range(n-i+1):
            print(end=" ")
            if (i==0 or j==0):
                coeff=1
            else:
                coeff=(coeff*(i-j+1))/j
```

Output: Enter the no. of rows: 4

1	1	1	1
1	2	1	
1	3	3	1

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF

: CSE

NAME OF THE LABORATORY : PP LAB

Name K. Sree Indira Sivani Roll No. 1602-21-733-052 Page No. 48

```
print (coeff, end=" ")
print ()
m = int(input("Enter the number of rows"))
pascal(m)
```

Create a user-defined module with functions to find
(i) smallest number ; (ii) largest number. Import the module
into another program:

* mod.py:

```
def small(a,b,c):
    if a < b and a < c:
        print (a, "is smallest")
    elif b < a and b < c:
        print (b, "is smallest")
    else:
        print (c, "is smallest")

def large(a,b,c):
    if a > b and a > c:
        print (a, "is largest")
    elif b > a and b > c:
        print (b, "is largest")
    else:
        print (c, "is largest")
```

VASAVI COLLEGE OF ENGINEERING

(AUTONOMOUS)
(Affiliated to Osmania University)
Hyderabad - 500 031.

DEPARTMENT OF : CSE

NAME OF THE LABORATORY : PP LAB

Name: K'Sree Indira Sivani Roll No. 1602-21-733-052 Page No. 44

```
import mod
print("Enter 3 numbers")
x=int(input())
y=int(input())
z=int(input())
mod.small(x,y,z)
mod.large(x,y,z).
```

Output:
Enter 3 numbers
5
10
15
5 is smallest.
15 is largest.

WAP to add the given numbers which are given as command line arguments:

```
import sys
s="0"
for i in range(1, len(sys.argv)):
    s=s+sys.argv[i]
print("Sum=", s)
```

Output:
Sum=01hkl856.

WAP to implement towers of hanoi: