EXPERIMENT NO-10

- Aim - Program to find the length of a list using recursion.

- Theory -

Recursion -

i) Recursion in python is a powerful technique that allows a function to call itself.
ii) It's like a loop, but instead of using iterative statement, we use recursive calls.
iii) Recursion is commonly used for solving problems that can be divided into smaller, similar subproblem.
iv) However, its important to ensure that the recursive call eventually reach the base case to avoid infinite recursion.

For example -

```
def factorial(n):
    if n==0:
        return 1
    else:
        return n* factorial(n-1)

result = factorial(5)
print("Factorial of 5 is", result)
```

Explaination -

i) In this example, the "factorial()" demonstrates the concept of recursion by breaking down the problem of calculating the factorial of a number into smaller instances of the same problem until a base case is reached.

- Program -

```
def calculate(my_list):
    if my_list == []:
        return 0
    else:
        return 1 + calculate(my_list[1:])

my_list = ["Priya", 2, "Tanu", "Aditya", 45, "Apple"]
length_of_list = calculate(my_list)
print("Length is:", length_of_list)
```

// Output -

Length is :- 6

- Explaination -

i) The 'calculate()' function takes a list 'my_list' as input.

ii) The if condition checks if the list is empty

iii) If it is, it return 0, indicated that the length of the empty list is 0.
iv) If the list is not empty, the function uses recursion
v) It has one to the length of the list obtained by slicing of the first element of 'my_list' ('my_list [1:]')
vi) This effectively reduces the problem cells by 1 element to continues until the base case is reach.
vii) Each recursive call reduces the size of the list by 1 element, and the sum of this increments gives the total length of the list.

- Conclusion -

i) This program demonstrates the use of recursion to find the length of a list in python.
ii) Recursion is a powerful technique that can simplify the implementation of certain algorithms, but it can also lead to stack overflow errors if not used carefully.
iii) In this case the program uses recursion to elegantly handle the task of calculating the length of a list, showcasing the simplicity and elegance of recursive solutions when applied appropriately.