

# Bahria University,

## Karachi Campus



### LAB EXPERIMENT NO. \_06\_ LIST OF TASKS

TASK NO	OBJECTIVE
Task 1	Write a program to calculate factorial of any given number by using recursion.
Task 2	Write a program to calculate the Fibonacci series of any given number using recursion.
Task 3	Implementation of Tower of Hanoi problem using recursion.
Task 4	N/A
Task 5	N/A
Task 6	N/A
Task 7	N/A
Task 8	N/A

**Submitted On:**

\_\_08/05/2020\_\_  
(Date: DD/MM)

**Task No. 1:** Write a program to calculate factorial of any given number by using recursion.

**Coding:**

```
[*] Task1.cpp Task2.cpp Task3.cpp
1  #include <iostream>
2  using namespace std;
3  int Factorial(int num)
4  {
5      if (num <= 1)
6      { return 1; }
7      else
8      { return num * Factorial(num - 1); }
9  }
10 int main()
11 {
12     int num;
13     cout << "Enter Number For Factorial = ";
14     cin >> num;
15     cout << "Factorial Of " << num << " is = " << Factorial(num);
16 }
```

**Output:**

```
E:\4th semester\Data Structure and Algorithms\Recursion\Task1.exe
Enter Number For Factorial = 6
Factorial Of 6 is = 720
-----
Process exited after 2.351 seconds with return value 0
Press any key to continue . . .
```

**Task No. 2:** Write a program to calculate the Fibonacci series of any given number using recursion.

### Coding:

```
Task1.cpp Task2.cpp Task3.cpp
1  #include <iostream>
2  using namespace std;
3  int fibonnaci(int x) {
4      if ((x == 1) || (x == 0)) {
5          return(x);
6      }
7      else {
8          return(fibonnaci(x - 1) + fibonnaci(x - 2));
9      }
10 }
11 int main() {
12     int x;
13     cout << "Enter Any Number For Fibonnaci series = ";
14     cin >> x;
15     cout << "Fibonnaci Series Will be "<<endl;
16     for (int i=0;i < x;i++)
17     {
18         cout << " " << fibonnaci(i);
19     }
20     return 0;
21 }
```

### Output:

```
E:\4th semister\Data Strcture and Algorithms\Recursion\Task2.exe
Enter Any Number For Fibonnaci series = 9
Fibonnaci Series Will be
0 1 1 2 3 5 8 13 21
-----
Process exited after 3.398 seconds with return value 0
Press any key to continue . . .
```

### Task No. 3: Implementation of Tower of Hanoi problem using recursion.

#### Coding:

```
Task1.cpp Task2.cpp [*] Task3.cpp
1  #include <iostream>
2  using namespace std;
3  int moves(0);
4  void Hanoi(int m, char a, char b, char c);
5
6  void Hanoi(int m, char a, char b, char c) {
7      moves++;
8      if (m == 1) {
9          cout << "Move disc " << m << " from " << a << " to " << c << endl;
10     }
11     else {
12         Hanoi(m - 1, a, c, b);
13         cout << "Move disc " << m << " from " << a << " to " << c << endl;
14         Hanoi(m - 1, b, a, c);
15     }
16 }
17 int main()
18 {
19     /* What program will do ?
20     move disc 1 A to c ACB
21     move disc 2 A to b ABC
22     move disc 1 C to b CBA
23     move disc 3 A to c ACB
24     move disc 1 B to a BAC
25     move disc 2 B to c BCA
26     move disc 1 A to c ACB */
27     int discs;
28     cout << "Enter the number of discs = ";
29     cin >> discs;
30     Hanoi(discs, 'A', 'B', 'C');
31     cout << "It took " << moves << " moves." << endl;
32     system("pause");
}
```

#### Output:

```
E:\4th semester\Data Structure and Algorithms\Recursion\Task 3.exe
Enter the number of discs = 2
Move disc 1 from A to B
Move disc 2 from A to C
Move disc 1 from B to C
It took 3 moves.
Press any key to continue . . .
```