

Assignment 4 Solutions

1. Write a Python Program to find the factorial of a number ?

```
In [1]: def factorial(num):  
        if (num < 1):  
            return 1  
        else:  
            return num*factorial(num-1)  
num = int(input('Enter a number: '))  
value = factorial(num)  
print(f'The Factorial of {num} is {value}')
```

Enter a number: 6
The Factorial of 6 is 720

2. Write a Python Program to display the multiplication table ?

```
In [2]: def generateTable(base,entries):  
        for x in range(1,entries+1):  
            print(f'{base} X {x} = {base*x}')  
  
num = int(input('Enter a number: '))  
values = int(input('Enter no of entries: '))  
generateTable(num,values)
```

Enter a number: 12
Enter no of entries: 12
12 X 1 = 12
12 X 2 = 24
12 X 3 = 36
12 X 4 = 48
12 X 5 = 60
12 X 6 = 72
12 X 7 = 84
12 X 8 = 96
12 X 9 = 108
12 X 10 = 120
12 X 11 = 132
12 X 12 = 144

3. Write a Python Program to print the fibonacci sequence ?

```
In [11]: s_count = int(input('Enter the no of fibonacci sequences you want? '))  
initial_list = [0,1]  
if s_count < 0:  
    print('Fibonacci Numbers are not available for Negative Numbers')  
elif s_count <= 2 and s_count >= 0:  
    print(initial_list)  
else:  
    for ins in range(s_count):  
        if ins >= 2:  
            initial_list.append(initial_list[ins-1]+initial_list[ins-2])  
    print(f'The First {s_count} fibonacci series are: ',initial_list)
```

Enter the no of fibonacci sequences you want? 30
The First 30 fibonacci series are: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811, 514229]

4. Write a Python Program to check Armstrong number ?

```
In [1]: def checkArmstrongNumber():  
        in_num = input('Enter a number: ')  
        sum = 0  
        for char in range(len(in_num)):  
            sum = sum + pow(int(in_num[char]),3)  
        if sum == int(in_num):  
            print(f'{in_num} is a Armstrong Number')  
        else:
```

```
print(f'{in_num} is a Not Armstrong Number')

for x in range(2):
    checkArmstrongNumber()
```

Enter a number: 100
100 is a Not Armstrong Number
Enter a number: 153
153 is a Armstrong Number

5. Write a Python Program to Find Armstrong number in an interval ?

```
In [2]: def checkArmstrongNumber(in_num, storage):
        sum = 0
        for char in range(len(in_num)):
            sum = sum + pow(int(in_num[char]),3)
        if sum == int(in_num):
            storage.append(int(in_num))

        start_interval = int(input('Enter the start of the Interval: '))
        end_interval = int(input('Enter the End of the Interval: '))
        list_of_armstrong = []

        if start_interval > end_interval:
            print("Start Interval Cannot be Greater than Interval:")
        else:
            for number in range(start_interval, end_interval+2):
                checkArmstrongNumber(str(number), list_of_armstrong)
            print(f'The Armstrong numbers between {start_interval} and {end_interval} are {list_of_armstrong}')
```

Enter the start of the Interval: 2
Enter the End of the Interval: 20000
The Armstrong numbers between 2 and 20000 are [153, 370, 371, 407]

6. Write a Python Program to sum of natural numbers ?

```
In [5]: def sumOfNaturalNumbers(num):
        sum = num*(num+1)/2
        print(f'Sum of {num} natural number is {sum}')

        num = int(input('Enter a number: '))
        sumOfNaturalNumbers(num)
```

Enter a number: 200
Sum of 200 natural number is 20100.0

In []:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js