

**Assignment -4**  
**Python Programming**

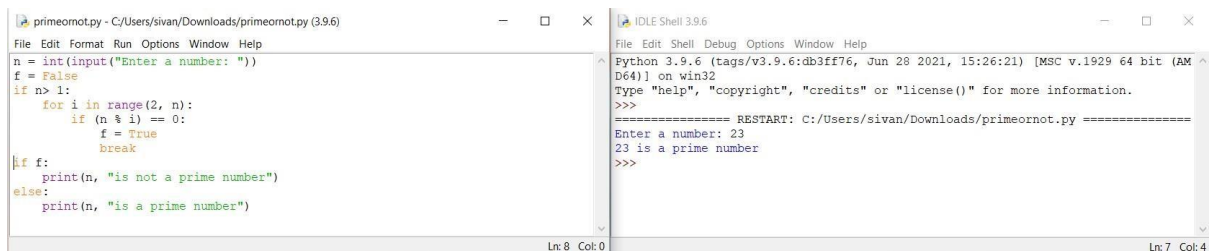
Assignment Date	19 September 2022
Student Name	Sathish Kumar G S
Team ID	PNT2022TMID00767
Maximum Marks	2 Marks

**Question-1:**

Write a python program to test a given number is prime or not.

**Solution :**

```
n = int(input("Enter a
number: "))
f = False
if n > 1:
    for i in range(2, n):
        if (n % i) == 0:
            f = True
            break
if f:
    print(n, "is not a prime
number")
else:
    print(n, "is a prime
number")
```



The screenshot shows a Python IDE with two windows. The left window displays the source code for a program to check if a number is prime. The code uses a for loop to test divisibility from 2 to n-1. If a divisor is found, it sets a flag 'f' to True and breaks the loop. If 'f' is True after the loop, it prints that the number is not prime; otherwise, it prints that the number is prime. The right window shows the execution of the program. It prompts the user to enter a number, and the user enters 23. The program outputs '23 is a prime number'.

```
primeornot.py - C:/Users/sivan/Downloads/primeornot.py (3.9.6)
File Edit Format Run Options Window Help
n = int(input("Enter a number: "))
f = False
if n > 1:
    for i in range(2, n):
        if (n % i) == 0:
            f = True
            break
if f:
    print(n, "is not a prime number")
else:
    print(n, "is a prime number")
Ln: 8 Col: 0

IDLE Shell 3.9.6
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/sivan/Downloads/primeornot.py =====
Enter a number: 23
23 is a prime number
>>>
```

**Question-2:**

Write a program to generate odd numbers from m to n using while loop

**Solution :**

```
min = int(input(" Enter any min value:"))
max = int(input(" Enter any max Value : "))
X=1;
if (min < max):
```

```

while X in range(min,max+ 1):
    if( X % 2 != 0):
        print("{0}".format(X))
        X=X+1;
else:
    print("min value you've entered is greater than max value")

```

The screenshot shows the Python IDLE Shell interface. The left pane displays the code for `odd.py`, which prompts the user for a minimum and maximum value and prints odd numbers in that range. The right pane shows the execution output, where the user has entered 1 for the minimum and 5 for the maximum, resulting in the output: 1, 3, 5.

### Question-3:

Write a Python program to display prime number series upto given number.

**Solution :**

```

l = 1
u = int(input("Enter the number : "))
print("Prime numbers between",l,"and",u,"are:")
for n in range(l,u + 1):
    if n > 1:
        for i in range(2, n):
            if (n%i) == 0:
                break
        else:
            print(n)

```

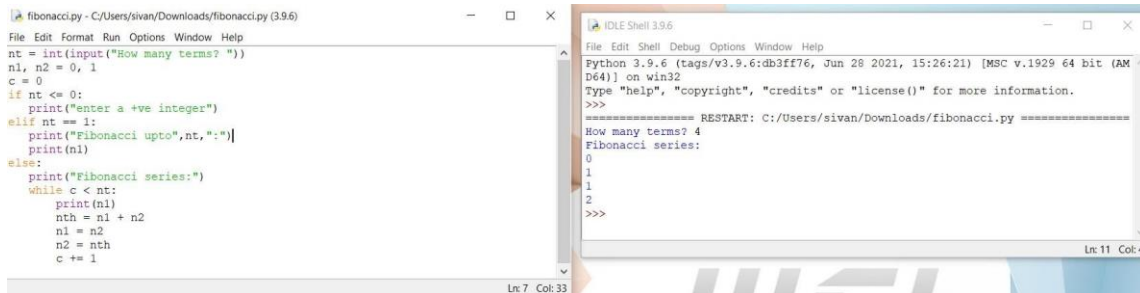
The screenshot shows the Python IDLE Shell interface. The left pane displays the code for `primenumberrange.py`, which prompts the user for a number and prints all prime numbers between 1 and that number. The right pane shows the execution output, where the user has entered 10, resulting in the output: Prime numbers between 1 and 10 are: 2, 3, 5, 7.

#### Question-4:

Write a Python program to generate Fibonacci series.

**Solution :**

```
nt = int(input("How many terms? "))
n1, n2 = 0, 1
c = 0
if nt <= 0:
    print("enter a +ve integer")
elif nt == 1:
    print("Fibonacci upto",nt,":")
    print(n1)
else:
    print("Fibonacci series:")
    while c < nt:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        c += 1
```



The screenshot displays two windows from a Python IDE. The left window, titled 'fibonacci.py - C:/Users/sivan/Downloads/fibonacci.py (3.9.6)', shows the source code of the program. The right window, titled 'IDLE Shell 3.9.6', shows the execution output. The user has entered '4' for the number of terms, and the program has printed the first four terms of the Fibonacci series: 0, 1, 1, 2.

```
fibonacci.py - C:/Users/sivan/Downloads/fibonacci.py (3.9.6)
File Edit Format Run Options Window Help
nt = int(input("How many terms? "))
n1, n2 = 0, 1
c = 0
if nt <= 0:
    print("enter a +ve integer")
elif nt == 1:
    print("Fibonacci upto",nt,":")
    print(n1)
else:
    print("Fibonacci series:")
    while c < nt:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        c += 1

IDLE Shell 3.9.6
File Edit Shell Debug Options Window Help
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/sivan/Downloads/fibonacci.py =====
How many terms? 4
Fibonacci series:
0
1
1
2
>>>
```