

Sri Lanka Institute of Information Technology



Lab Submission
<Lab sheet 03>

<IT24101537>

<S. Anojan>

Discrete Mathematics | IT1160

B.Sc. (Hons) in Information Technology

Part 01.

Q1,2,3)

jupyter Untitled1 Last Checkpoint: 12 minutes ago

File Edit View Run Kernel Settings Help

📁 + ✂ 📄 📋 ▶ ■ ↺ ▶▶ Code ▼

```
[5]: num = int(input("Enter a number: "))
    if num % 2 == 0:
        print("Even Number")
    else:
        print("Odd Number")
```

Enter a number: 2
Even Number

```
[11]: N = int(input("Enter a number: "))
    sum_natural = 0
    for i in range(1, N + 1):
        sum_natural += i
    print(sum_natural)
```

Enter a number: 5
15

```
[15]: numbers = []
    for i in range(5):
        num = int(input(f"Enter number {i+1}: "))
        numbers.append(num)
    print(numbers)
```

Enter number 1: 1
Enter number 2: 2
Enter number 3: 3
Enter number 4: 4
Enter number 5: 5
[1, 2, 3, 4, 5]

```
[ ]:
```

Q4,5)

jupyter Untitled1 Last Checkpoint: 18 minutes ago

File Edit View Run Kernel Settings Help

📄 + ✂ 📄 📄 ▶ ■ ↺ ▶▶ Code ▾

```
[17]: num = int(input("Enter a number: "))
      factorial = 1
      i = num
      while i > 0:
          factorial *= i
          i -= 1
      print(factorial)
```

Enter a number: 5
120

```
[19]: #a)
      print("First 10 terms of the sequence:")
      for n in range(1, 11):
          an = n**2 + 3*n
          print(an, end=" ")
      print()
```

First 10 terms of the sequence:
4 10 18 28 40 54 70 88 108 130

```
[21]: #b)
      a = 5 # First term
      d = 4 # Common difference
      n = 10 # Number of terms
      Sn = (n / 2) * (2 * a + (n - 1) * d)
      print("Sn =", Sn)
```

Sn = 230.0

[]: 🌟 📄 ⬆ ⬇ 📄 🗑

Part 02.

Q6,7)

jupyter Untitled2 Last Checkpoint: 14 minutes ago

File Edit View Run Kernel Settings Help

Enter the number : 12345
54321

```
[9]: num = input("Enter the number :")
      reversed_num = num[::-1]
      print(reversed_num)
```

Enter the Number : 123456
The total count is : 6

```
[13]: num = int(input("Enter the Number :"))
      count = 0
      while num > 0:
          num //= 10
          count += 1
      print("The total count is :",count)
```

Q8)

jupyter Untitled2 Last Checkpoint: 15 minutes ago

File Edit View Run Kernel Settings Help

```
[19]: num = []

      for x in range(7):
          number = int(input(f"Enter the Number (i+1) :"))
          num.append(number)

      largest_num = num[0]
      smallest_num = num[0]


      for x in range(7):
          if(largest_num < num[x]):
              largest_num = num[x]
          elif(smallest_num > num[x]):
              smallest_num = num[x]

      print("The largest Number is :",largest_num)
      print("The smallest Number is :",smallest_num)
```

Enter the Number (i+1) : 1
Enter the Number (i+1) : 2
Enter the Number (i+1) : 3
Enter the Number (i+1) : 4
Enter the Number (i+1) : 5
Enter the Number (i+1) : 6
Enter the Number (i+1) : 7
The largest Number is : 7
The smallest Number is : 1

[]:

Q9,10)

 jupyter Untitled2 Last Checkpoint: 30 minutes ago

File Edit View Run Kernel Settings Help

 +        Code ▾

```
[27]: def is_prime(num):
      if num <= 1:
          return False
      for x in range(2, num):
          if num % x == 0:
              return False
          return True
      for number in range(1, 21):
          if is_prime(number):
              print(number, end=" ")
```

3 5 7 9 11 13 15 17 19

```
[37]: num1 = []
      for x in range(7):
          num = int(input(f"Enter a Number {x+1} :"))
          num1.append(num)

      num1.sort(reverse=True)
      print(num1)
      second_largest = num1[1]
      print("The second largest Number is :",second_largest)
```

```
Enter a Number 1 : 1
Enter a Number 2 : 2
Enter a Number 3 : 3
Enter a Number 4 : 4
Enter a Number 5 : 5
Enter a Number 6 : 6
Enter a Number 7 : 7
[7, 6, 5, 4, 3, 2, 1]
The second largest Number is : 6
```

Q11,12)

jupyter Untitled2 Last Checkpoint: 55 minutes ago

File Edit View Run Kernel Settings Help

📁 + ✂ 📄 📌 ▶ ■ ↺ ▶▶ Code ▼

```
[39]: num = int(input("Enter the Number :"))
      factors = []
```

```
      for x in range(1, num + 1):
          if num % x == 0:
              factors.append(x)

      print("Factors of", num, "are", factors)
```

```
Enter the Number : 12
Factors of 12 are [1, 2, 3, 4, 6, 12]
```

```
[47]: def gcd(num1, num2):
      while num2 != 0:
          num1, num2 = num2, num1 % num2
      return num1

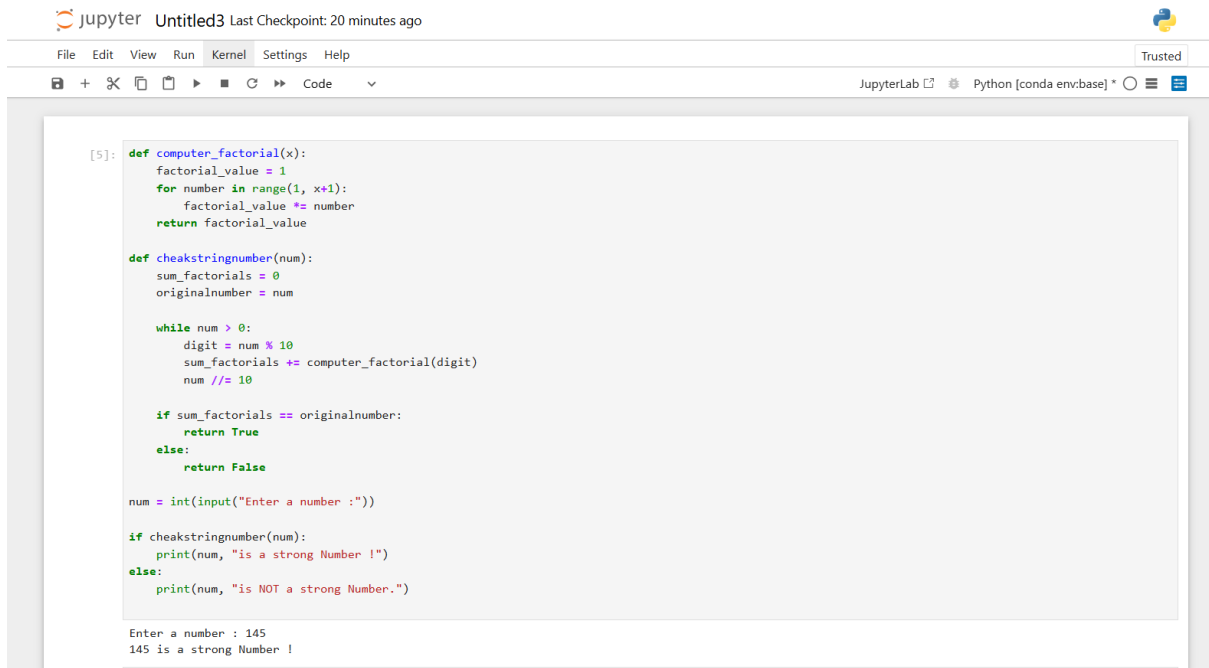
      num_1 = int(input("Enter the 1st Number :"))
      num_2 = int(input("Enter the 2st Number :"))

      result = gcd(num_1, num_2)
      print("The Gcd of", num_1, "and", num_2, "is", result)
```

```
Enter the 1st Number : 10
Enter the 2st Number : 30
The Gcd of 10 and 30 is 10
```

```
[ ]:
```

Q13)



JupyterLab interface showing a Python script for checking strong numbers. The script defines two functions: `computer_factorial(x)` and `cheakstringnumber(num)`. The `computer_factorial` function calculates the factorial of a given number. The `cheakstringnumber` function checks if a number is a strong number by summing the factorials of its digits and comparing the result to the original number. The script prompts the user to enter a number and prints the result.

```
[5]: def computer_factorial(x):
    factorial_value = 1
    for number in range(1, x+1):
        factorial_value *= number
    return factorial_value

def cheakstringnumber(num):
    sum_factorials = 0
    originalnumber = num

    while num > 0:
        digit = num % 10
        sum_factorials += computer_factorial(digit)
        num //= 10

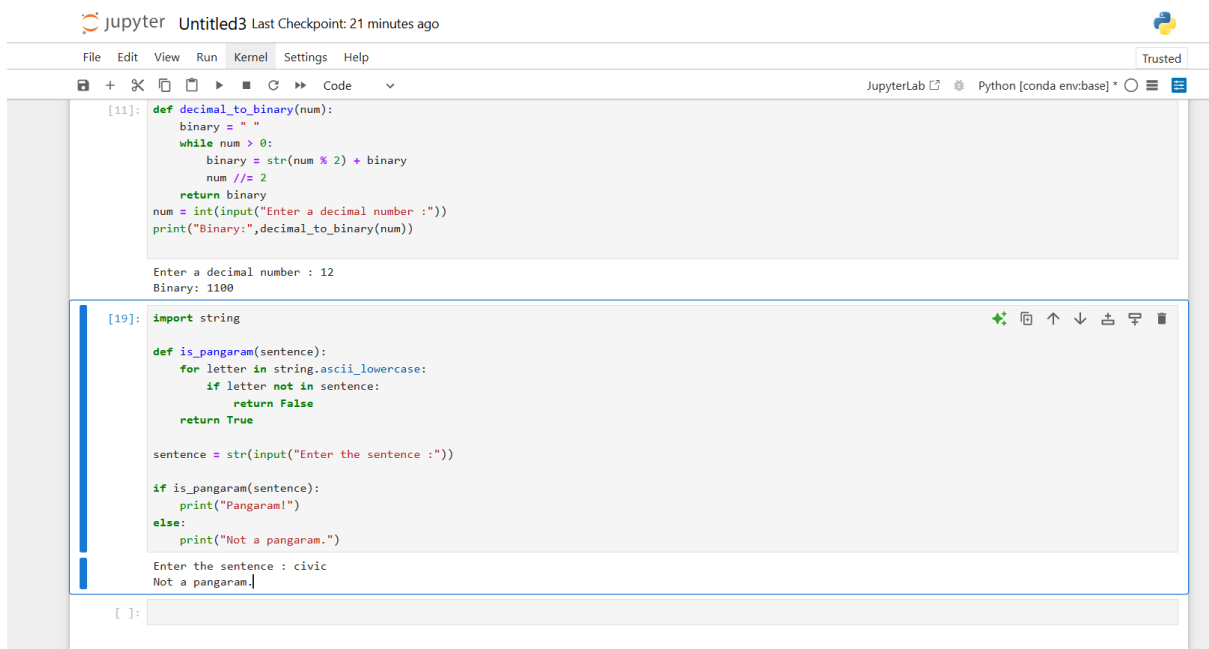
    if sum_factorials == originalnumber:
        return True
    else:
        return False

num = int(input("Enter a number :"))

if cheakstringnumber(num):
    print(num, "is a strong Number !")
else:
    print(num, "is NOT a strong Number.")

Enter a number : 145
145 is a strong Number !
```

Q14,15)



JupyterLab interface showing two Python scripts. The first script, `decimal_to_binary(num)`, converts a decimal number to its binary representation. The second script, `is_pangram(sentence)`, checks if a sentence is a pangram (contains all letters of the alphabet). The script prompts the user to enter a decimal number and a sentence, and prints the results.

```
[11]: def decimal_to_binary(num):
    binary = " "
    while num > 0:
        binary = str(num % 2) + binary
        num //= 2
    return binary
num = int(input("Enter a decimal number :"))
print("Binary:", decimal_to_binary(num))

Enter a decimal number : 12
Binary: 1100

[19]: import string

def is_pangram(sentence):
    for letter in string.ascii_lowercase:
        if letter not in sentence:
            return False
    return True

sentence = str(input("Enter the sentence :"))

if is_pangram(sentence):
    print("Pangram!")
else:
    print("Not a pangram.")

Enter the sentence : civic
Not a pangram.
```