Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**5**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **1** | Write a python program that can find value of tan(0.12) using Newton’s forward interpolation formula, with considering to the following data points: |
| 2 | Write a python program implementing the backward interpolation formula that considers the following data points, and   1. Find the value of y at x = 300 2. Find the value of y at any given user input |

Submitted On:

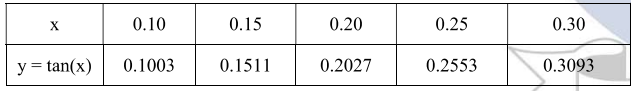
22 October 2024

\_\_\_\_\_\_\_\_\_\_\_\_

(Date: DD/MM/YY)

**Task 1**

Write a python program that can find value of tan (0.12) using Newton’s forward interpolation formula, with considering to the following data points:



**Solution:**

import numpy as np

x\_values = np.array([0.10, 0.15, 0.20, 0.25, 0.30])

y\_values = np.array([0.1003, 0.1511, 0.2027, 0.2553, 0.3093])

n = len(x\_values)

difference\_table = np.zeros((n, n))

difference\_table[:, 0] = y\_values

for j in range(1, n):

    for i in range(n - j):

        difference\_table[i, j] = difference\_table[i + 1, j - 1] - difference\_table[i, j - 1]

print("Forward Difference Table:")

print(difference\_table)

x = 0.12

h = x\_values[1] - x\_values[0]

p = (x - x\_values[0]) / h

interpolated\_value = y\_values[0]

product\_term = 1

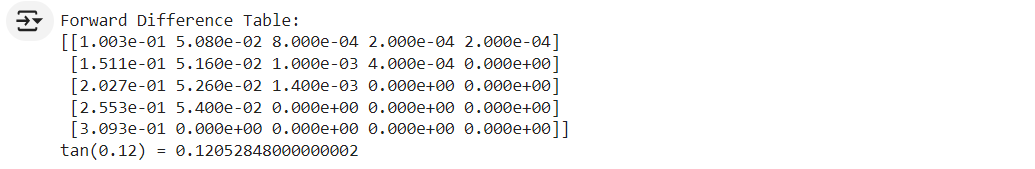
for i in range(1, n):

    product\_term \*= (p - (i - 1)) / i

    interpolated\_value += product\_term \* difference\_table[0, i]

print(f"tan(0.12) = {interpolated\_value}")

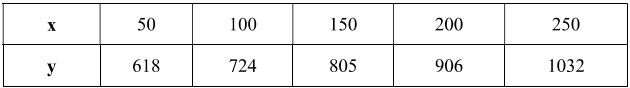
**Output:**

****

**Task 2**

**Write a python program implementing the backward interpolation formula that considers the following data points, and**

1. **Find the value of y at x = 300**
2. **Find the value of y at any given user input**



**Solution:**

import numpy as np

x\_values = np.array([50, 100, 150, 200, 250])

y\_values = np.array([618, 724, 805, 906, 1032])

n = len(x\_values)

difference\_table = np.zeros((n, n))

difference\_table[:, 0] = y\_values

for j in range(1, n):

    for i in range(n - 1, j - 2, -1):

        difference\_table[i, j] = difference\_table[i, j - 1] - difference\_table[i - 1, j - 1]

print("Backward Difference Table:")

print(difference\_table)

def newtons\_backward\_interpolation(x, x\_values, y\_values, difference\_table):

    h = x\_values[1] - x\_values[0]

    p = (x - x\_values[-1]) / h

    interpolated\_value = y\_values[-1]

    product\_term = 1

    for i in range(1, n):

        product\_term \*= (p + (i - 1)) / i

        interpolated\_value += product\_term \* difference\_table[-1, i]

    return interpolated\_value

x = 300

y\_at\_300 = newtons\_backward\_interpolation(x, x\_values, y\_values, difference\_table)

print(f"y at x = 300 is {y\_at\_300}")

user\_input = float(input("Enter the value of x: "))

y\_at\_user\_input = newtons\_backward\_interpolation(user\_input, x\_values, y\_values, difference\_table)

print(f"y at x = {user\_input} is {y\_at\_user\_input}")

**Output:**

****