Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

**6**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **1** | **Write a python program implementing the newton’s difference formula that considers the**  **following data points, and**  **a. Find the value of y at x = 301**  **b. Find the value of y at any given user input** |
| **2** | **Write a Python program that implements Newton's difference formula, reads the CSV file 'salary\_data,' and considers 'years of experience' as x and 'salary' as y. Then, upon user input for a specific number of years of experience, it provides an output of the salary they would receive.** |
| 3 | Write a python program implementing the Lagrange interpolation formula that considers the following data points, and find the value of y at x = 7 |

Submitted On:

**Date: 30/10/2023**

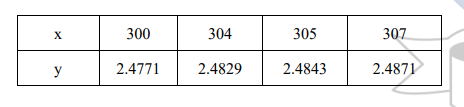
**Task No. 01:**

**Write a python program implementing the newton’s difference formula that considers the**

**following data points, and**

**a. Find the value of y at x = 301**

**b. Find the value of y at any given user input**



**Solution:**

def calculate\_product\_term(i, value, x):

product = 1

for j in range(i):

product \*= (value - x[j])

return product

def calculate\_divided\_difference\_table(x, y, n):

for i in range(1, n):

for j in range(n - i):

y[j][i] = ((y[j][i-1] - y[j+1][i-1]) / (x[j] - x[i+j]))

return y

def apply\_newton\_formula(value, x, y, n):

result = y[0][0]

for i in range(1, n):

result += (calculate\_product\_term(i, value, x) \* y[0][i])

return result

def display\_divided\_difference\_table(y, n):

for i in range(n):

for j in range(n - i):

print(round(y[i][j], 4), "\t", end=" ")

print("")

n = int(input("Enter the number of inputs: "))

x = []

y = [[0 for i in range(n)] for j in range(n)]

for i in range(n):

x\_val = float(input(f"Enter x[{i}]: "))

y\_val = float(input(f"Enter y[{i}]: "))

x.append(x\_val)

y[i][0] = y\_val

y = calculate\_divided\_difference\_table(x, y, n)

print("\nDivided Difference Table:")

display\_divided\_difference\_table(y, n)

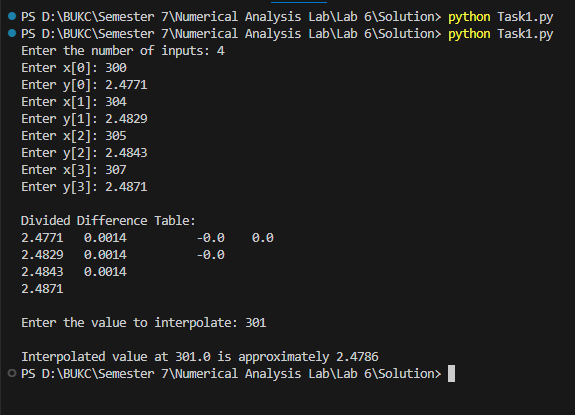
value = float(input("\nEnter the value to interpolate: "))

interpolated\_value = apply\_newton\_formula(value, x, y, n)

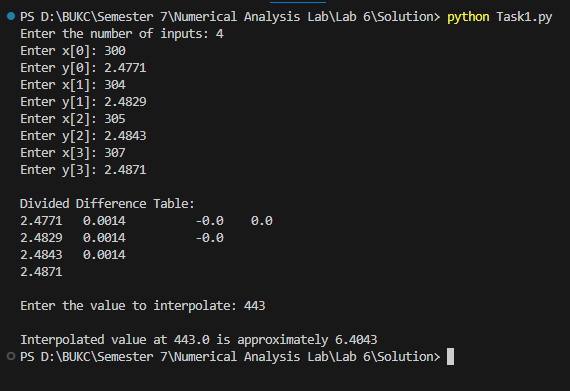
print("\nInterpolated value at", value, "is approximately", round(interpolated\_value, 4))

**Output:**

**value of y at x = 301:**



***value of y at any given user input***



**Task No. 02:**

**Write a Python program that implements Newton's difference formula, reads the CSV file 'salary\_data,' and considers 'years of experience' as x and 'salary' as y. Then, upon user input for a specific number of years of experience, it provides an output of the salary they would receive.**

**Solution:**

import pandas

def calculate\_product\_term(i, value, x):

product = 1

for j in range(i):

product \*= (value - x[j])

return product

def calculate\_divided\_difference\_table(x, y, n):

for i in range(1, n):

for j in range(n - i):

y[j][i] = ((y[j][i-1] - y[j+1][i-1]) / (x[j] - x[i+j]))

return y

def apply\_newton\_formula(value, x, y, n):

result = y[0][0]

for i in range(1, n):

result += (calculate\_product\_term(i, value, x) \* y[0][i])

return result

df=pandas.read\_csv('./Data Set/salary-data-18102023-113523am.csv')

n = len(df)

x = df['YearsExperience']

y = [[0 for i in range(n)] for j in range(n)]

for i in range(n):

y[i][0] = df['Salary'][i]

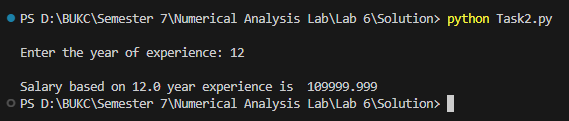
y = calculate\_divided\_difference\_table(x, y, n)

value = float(input("\nEnter the year of experience: "))

interpolated\_value = apply\_newton\_formula(value, x, y, n)

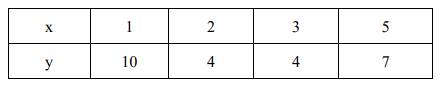
print("\nSalary based on", value, "year experience is ", round(interpolated\_value, 4))

**Output:**



**Task No. 03:**

**Write a python program implementing the Lagrange interpolation formula that considers the following data points, and find the value of y at x = 7**



**Solution:**

class DataPoint:

def \_\_init\_\_(self, x, y):

self.x = x

self.y = y

def lagrange\_interpolation(data\_points, x\_value):

result = 0.0

for i in range(len(data\_points)):

term = data\_points[i].y

for j in range(len(data\_points)):

if j != i:

term \*= (x\_value - data\_points[j].x) / (data\_points[i].x - data\_points[j].x)

result += term

return result

if \_\_name\_\_ == "\_\_main\_\_":

data\_points = []

n = int(input("Enter the number of known data points: "))

for i in range(n):

x = float(input(f"Enter x{i + 1}: "))

y = float(input(f"Enter y{i + 1}: "))

data\_points.append(DataPoint(x, y))

x\_value = float(input("Enter the x value for interpolation: "))

interpolated\_value = lagrange\_interpolation(data\_points, x\_value)

print(f"Interpolated value at {x\_value} is: {interpolated\_value}")

**Output:**

