# Department of Computing

# MATH 333: Numerical Analysis

# Class: BSCS-9ABC

# Lab 10: Derivatives using Newton’s Forward, Backward Difference Interpolation Formula and Stirling’s Central Difference Interpolation Formula

# Date: April 15, 2022

# Time: 10:00 - 1:00 & 2:00 – 5:00

# Lab Engineer: Anum Asif

|  |
| --- |
| **SUBMITTED BY:**  **Fatima Seemab**  **291310**  **Lab 10** |

# Lab 10: Derivatives using Newton’s Forward, Backward Difference Interpolation Formula and Stirling’s Central Difference Interpolation Formula

**Introduction**

Matlab represents polynomials with a vector of coefficients. The length of the vector will always be one more than the order of the polynomial

y = x2 + 5x -3 is represented as [1 5 -3]

y = -x3/3 + x is represented as [-1/3 0 1 0]

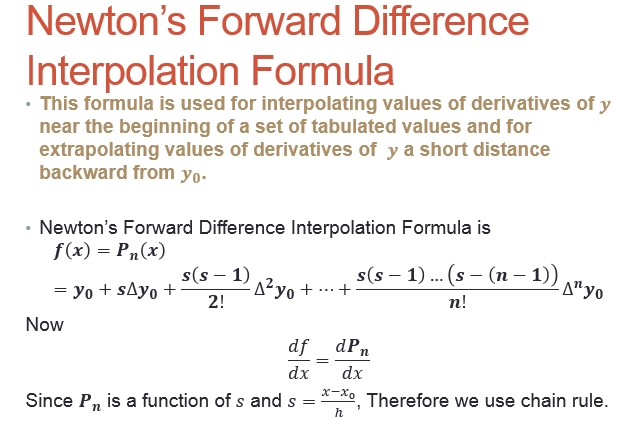
**Objectives**

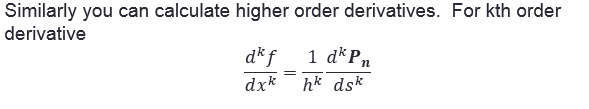
The purpose of this lab is to get familiar with Derivatives using Different Interpolation method and online learning.

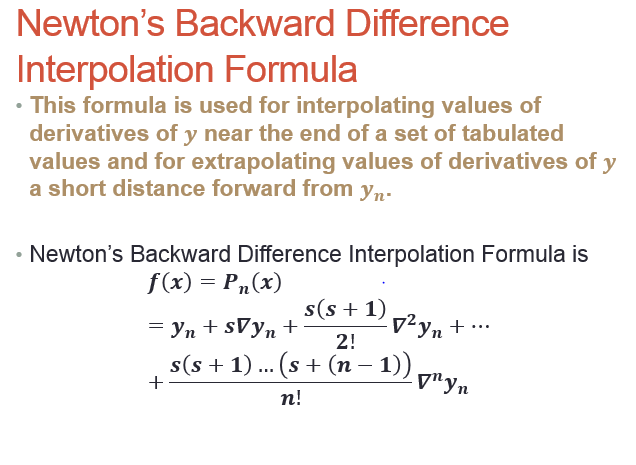
**Tools/Software Requirement**

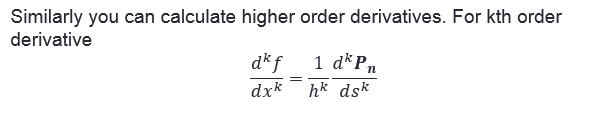
Matlab R2016a, MS 365, Matlab online

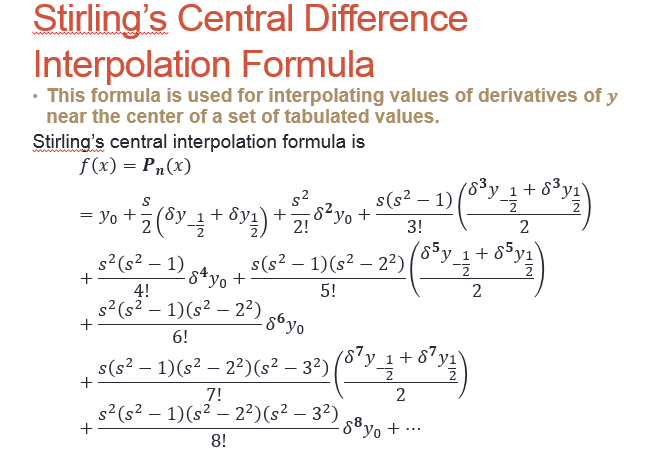
**Description**

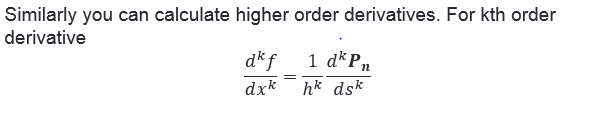












**NOTE: For formula’s Kindly check your lecture slides.**

**For** Stirling’s central difference formula, you can take help from the following link.

**http://matlabcode.weebly.com/numerical-methods-matlab-programming-examples/matlab-code-stirlings-interpolation-formula-numerical-methods**

**Steps:**

1. Make function for each interpolation or make a single function for all interpolation for a single data. (Or you can simply implement both tasks in a script file.
2. Functions will include the implementation of formula’s given in lecture slides and lab manual.
3. The input parameters will include input data given in lab tasks.
4. Take screen shots of your outputs.

**Lab Task:**

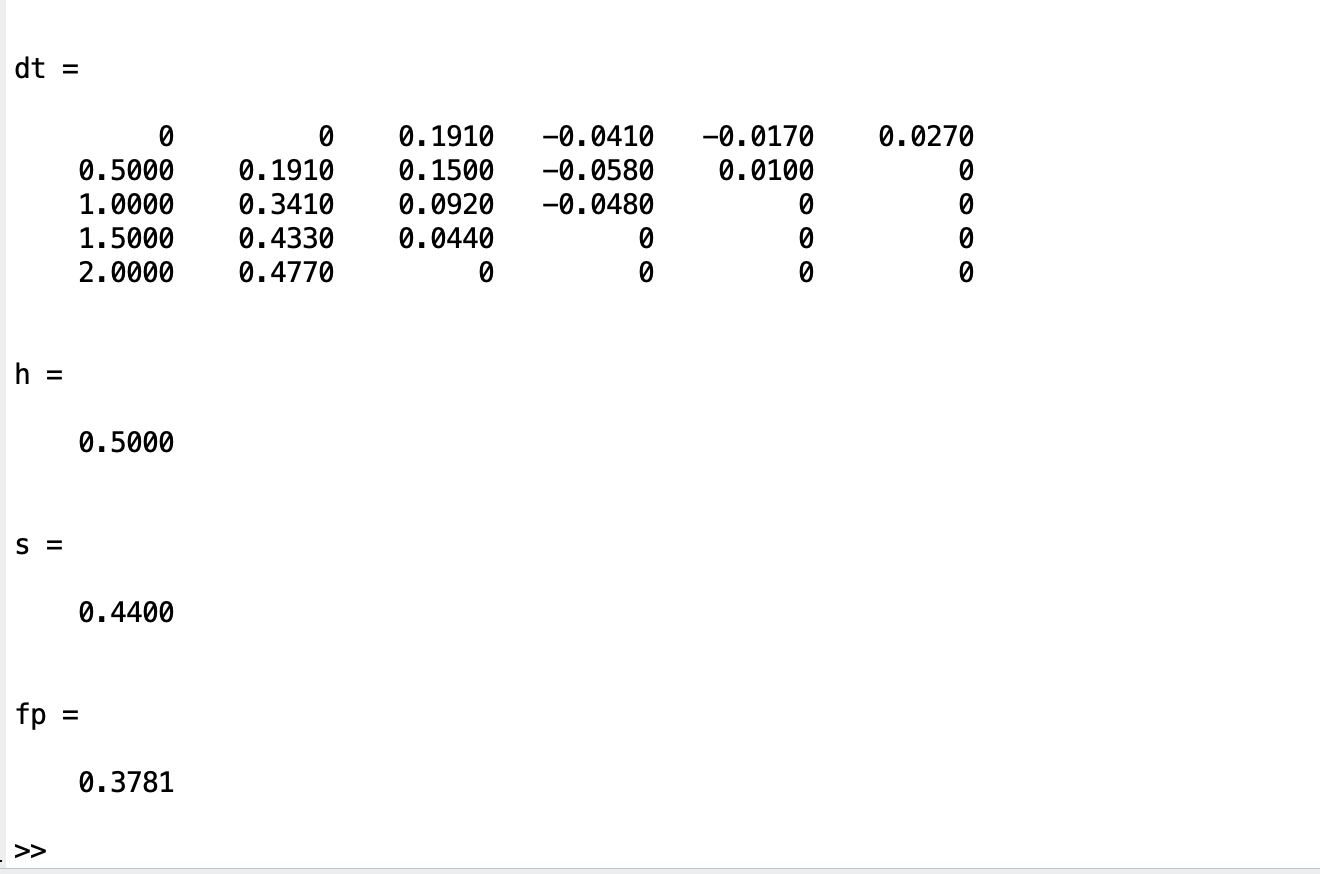
1. Given n no. of floating values x and their corresponding function values f(x), estimate the value of mathematical function at x=1.22, using Stirling’s central difference formula.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0.5 | 1.0 | 1.5 | 2.0 |
|  | 0 | 0.191 | 0.341 | 0.433 | 0.477 |

**CODE:**

|  |
| --- |
| X=[0 0.5 1.0 1.5 2.0]  y=[0 0.191 0.341 0.433 0.477];  dt=zeros(5,6);  for i=1:5  dt(i,1)=t(i);  dt(i,2)=o(i);  end  n=4;  for j=3:6  for i=1:n  dt(i,j)= dt((i+1),(j-1))-dt(i,(j-1))  end  n=n-1;  end  h=x(2)-x(1)  tp=1.22;  s=(tp-x(3))/h  l= tp-(s\*h);  p=s;  for i=1:5  if(l==t(i))  r=i;  end  end  f0=o(r);  f11=dt((r-1),3);  f12=dt((r+1),3);  f02=dt((r-1),4);  f31=dt((r-2),5);  f32=dt((r-1),5);  f04=dt((r-2),6);  fp=f0+(p\*(f11+f12))/2+((p\*p)\*f02)/2+(p\*((p\*p)-1)\*(f31+f32))/12+((p\*p)\*((p\*p)-1)\*f04)/24 |

**OUTPUT:**



1. Considering a uniform beam of one meter long simply supported at both ends, the bending moment is given by the following relation:

Where is the deflection, is the bending moment and is the flexural rigidity.

Calculate the bending moment at

(i): using Newton’s Forward Difference Formula

(ii): using Newton’s Backward Difference Formula,

assuming that the deflection distribution is among the following:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 |
| , | 0.0000000 | 7.7800000 | 10.6800000 | 8.3700000 | 3.9700000 | 0.000000 |

**CODE:**

|  |
| --- |
| clear  clc  x = [ 0.0 0.2 0.4 0.6 0.8 1.0];  y = [ 0.0 7.78 10.68 8.37 3.97 0.00];  xi = 0.25;  [result,yi,p] = newtint(x,y,xi,1)  yi = poly2sym(yi);  yi = diff(yi);  yi = diff(yi);  yi = inline(yi);  EI = 0.5;  value = yi(0.25)\*EI |

**OUTPUT:**

Graphical user interface, application

Description automatically generated

**TASK 2 Part 2**

**CODE:**

|  |
| --- |
| clear  clc  x = [ 0.0 0.2 0.4 0.6 0.8 1.0];  y = [ 0.0 7.78 10.68 8.37 3.97 0.00];  xi = 0.90;  [result,yi,p] = newtint(x,y,xi,1)  yi = poly2sym(yi);  yi = diff(yi);  yi = diff(yi);  yi = inline(yi);  EI = 0.5;  value = yi(0.90)\*EI |

**OUTPUT:**

Graphical user interface, text, application

Description automatically generated