Newton Backward Interpolation – Documentation

This program performs Newton Backward Interpolation to estimate the value of a function at a given point using backward differences of tabulated data.

OBJECTIVE

To calculate the interpolated value f(X) for a given value X from a set of discrete data points (xi, yi) using Newton Backward Interpolation formula.

THEORY

For n data points x0, x1, ..., x_{n-1} with corresponding function values y_0, y_1, ..., y_{n-1}, the backward interpolation formula is:

$$f(X) = yn + v\,\Delta yn + \left(\frac{v(v+1)}{2!}\right)\Delta^2 yn + \left(\frac{v(v+1)(v+2)}{3!}\right)\Delta^3 yn + \ldots$$

Where:

- $y_n = y_{n-1}$
- $v = \frac{(X - xn)}{h}$
- $h = x_i - x_{i-1}$
- $\Delta^k y_n$ = k-th backward difference at last point

INPUT FORMAT (input.txt)

T
n
x1 x2 ... xn
y1 y2 ... yn
X
(repeat for T test cases)

EXAMPLE INPUT

4
5
1 2 3 4 5

2 4 8 16 32
3

4
10 20 30 40
5 7 10 15
35

3
0 1 2
1 3 7
1

5
2 4 6 8 10
4 8 18 32 50
7

OUTPUT (Console + output.txt)

For each test case, the program prints:

1. Test Case Number
2. Number of data points (n)
3. x[] and y[] values
4. Interpolation point X
5. Full backward difference table (n×n including zeros)
6. Interpolated value at X

EXAMPLE OUTPUT

TestCase#1
n: 5
x: 1 2 3 4 5
y: 2 4 8 16 32
X: 3

Backward Difference Table:
2 0 0 0 0
4 2 0 0 0
8 4 2 0 0
16 8 4 2 0

32 16 8 4 2
Interpolated Value: 8

ALGORITHM

1. Read number of test cases T

    - For each test case:
      a. Read n, x[], y[], X
      b. Initialize n×n difference table with zeros
      c. Fill first column with y[] values
      d. Compute backward differences column by column
      e. Apply Newton Backward Formula to compute interpolated value
      f. Print inputs, difference table, and interpolated value to console and output file

FEATURES

- Handles multiple test cases
- Works with uniformly spaced data points
- Prints full backward difference table including zeros
- Displays all input and output for clarity
- Outputs results to both console and output.txt