Algorithm Problem Solving
17ECSE309

Horner’s Rule

USN: 01FE15BCS149
NAME: Rajesh S.

What problem does it solve?

- It helps in solving polynomial equations when the value of the variable is given
- Polynomial equation:
    - $P(X) = c_n x^n + c_{n-1} x^{n-1} + c_{n-2} x^{n-2} + \ldots + c_1 x + c_0$
        - Cn – Constant coefficients
        - X – variable
- Example: $P(X) = 2x^3 - 6x^2 + 2x - 1$ where $x = 3$ must give 5 as the solution

Horner’s Rule

- Taking the previous example - $P(X) = 2x^3 – 6x^2 + 2x – 1$ where $x = 3$
- By Horner’s rule we can solve it as:
  - Given coefficients: 2, -6, 2, -1 → (store in an array p)
  - Store p[0] in a variable result
  - Keeping the 1st element as initial and starting from 2nd element to last element of p

    multiply value of x with previous result and add current element

    i.e. result = result*x + p[i] → (where i iterates from 2nd index to last index of p)
  - $P(3) = (2*3 – 6) \rightarrow (0*3 + 2) \rightarrow (2*3 – 1)$

    $= 5$

C code – Horner’s rule function

```
int horner( int poly[] , int n, int x)
{
    int result = poly[0];  // Initialize result

    // Evaluate value of polynomial using Horner's method
    for ( int  I = 1; i < n; i++ )
        result = result*x + poly[i];
    return result;
} →[1]
```

Time complexity and Advantages

- Time Complexity: O(n)
- Problems with normal approach:
  - A naive way to evaluate a polynomial is to one by one evaluate all terms. First calculate x^n, multiply the value with c_n, repeat the same steps for other terms and return the sum. Time complexity of this approach is $O(n^2)$ if we use a simple loop for evaluation of x^n. Time complexity can be improved to O(nLogn) if we use [O(Logn) approach for evaluation of x^n].
- Why to go for Horner's rule:
  - As multiplication and addition operations are done only n times – we get a time complexity of O(n) instead of O(nlogn) and proves to have better time complexity compared to the improved naïve approach.

Applications

- To solve Newton's polynomial:
  - Example: Use Horner's rule to evaluate the Newton polynomial defined by the points **x** = $(0.5, 5.9, 1.3, 4.7, 3.5)^T$ with corresponding coefficients **c** = $(0.39, 0.47, 0.63, -0.53, 1.23)^T$ at the points x = 3.7 and x = 4.2. → [2]
- Extension of horner's method can be used in synthetic division of polynomials:
  - Example: Use Horner's method to solve – $(x^4 + 4x^3 + 3x^2 - 4x - 4) / (x - 1)$

  The result comes out to be $x^3 + 5x^2 + 8x + 4$ →[3]

References

[1] Horner’s Method Polynomial Evaluation, Link: https://www.geeksforgeeks.org/horners-method-polynomial-evaluation/

[2] Numerical Analysis, Link: https://ece.uwaterloo.ca/~dwharder/NumericalAnalysis/05Interpolation/horner/

[3] Introduction To Horner's Method Of Synthetic Division / Polynomials / Maths Algebra, Link: https://www.youtube.com/watch?v=3LjFgqDFxHQ