~~Q) Horner's Rule is a strategy for evaluating ...~~

- The full program is on my P.C.

```
1. long Horner_Rule(int coeff[], int n, int begin, int x_0)
2. {
3.   if(begin==n-1)
4.     return coeff[begin];
5.   else
6.     return coeff[begin]+x_0*Horner_Rule(coeff[], n, begin+1, x_0);
7. }
```

- coeff[] stores the coefficients of the polynomial, s.t. coeff[i] = a_i.
- begin indicates the beginning index of the current range of coeff[] to be evaluated, initial value 0.
- n: no. of coefficients, x_0: point of evaluation

Claim: The function Horner_Rule correctly computes the polynomial $a_n x^n + a_{n-1}x^{n-1} + \ldots\ldots + a_1 x_1 + a_0$ at the point x_0. The coefficients of the form a_i and x_0 is arbitrary.

Proof: The proof is by mathematical induction on n, the no. of coefficients.

Base Case: n=1. begin=0. ∵ begin=(n-1), we simply return the coefficient a_0 = coeff[0] on line 4. This correct.

Inductive hyp: Let, for an arbitrary K=n, Horner's rule correctly computes the polynomial $a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \ldots\ldots + a_1 x_1 + a_0$ at the point x_0. for arbitrary polynomial coefficients a_i and x_0.

Induction Step: Now, let K=n+1. Initially, begin=0. ∵ begin ≠ n−1 We move to line 6. By induction hypothesis, Horner_Rule(coeff, n+1, 1, x0) returns: $(a_n x_0^{n-1} + a_{n-1}x_0^{n-2} + \ldots\ldots + a_2 x_0 + a_1)$

We are returning: coeff[0] = $a_0 + x_0(a_n x_0^{n-1} + a_{n-1}x_0^{n-2} + \ldots\ldots + a_2 x_0 + a_1)$

$= \boxed{a_n x_0^n + a_{n-1}x_0^{n-1} + \ldots\ldots + a_2 x_0^2 + a_1 x_0 + a_0}$ which should be the case □