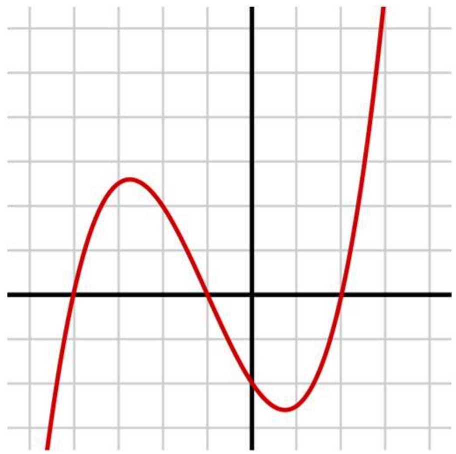
107transfer-bootstrap

B-MAT-200

Polynomial functions

- A polynomial is a function of the form:
$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$
- The constants a_i are the coefficients of the polynomial
- The degree of a polynomial is the highest power of x in its expression
- Example: $f(x) = \frac{1}{4}x^3 + \frac{3}{4}x^2 - \frac{3}{2}x - 2$

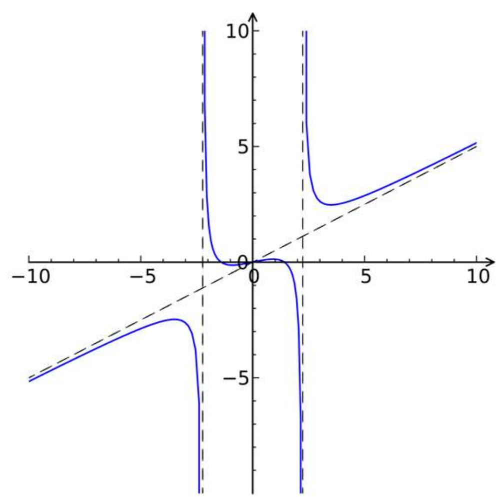


Rational functions

- A rational function is a function which can be defined as a fraction such as both numerator and denominator are polynomials:

$$f(x) = \frac{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0}{b_m x^m + b_{n-1} x^{n-1} + \cdots + b_2 x^2 + b_1 x + b_0}$$

- Example: $f(x) = \frac{x^3 - 2x}{2x^2 - 10}$



Polynomial evaluation – Horner's method

- We can rewrite the polynomial:

$$\begin{aligned} f(x) &= a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0 \\ &= \big((\cdots(a_n x + a_{n-1})x + \cdots + a_2)x + a_1\big)x + a_0 \end{aligned}$$

- We can then define the sequence:

$$\begin{aligned} p_n &= a_n \\ p_{n-1} &= a_{n-1} + p_n x \\ &\vdots \\ p_0 &= a_0 + p_1 x \end{aligned}$$

- Finally, we get $f(x) = p_0$, and we used only n additions and n multiplications

107transfer

- Goal: Evaluate rational functions for every values in a range from 0 to 1, with a step of 0.001
- Input: a list of strings representing numerators and denominators of rational functions

```
./107transfer num den [num den …]
```

- Output: display the function evaluation for every values between 0 and 1 with a step of 0.001
- Beware of division by 0
- Use types with enough precision

Inputs example

- Input format: strings that represent polynomial coefficients
  - “1*2*3*4” → $4x^3 + 3x^2 + 2x + 1$
- Examples
  - “1*2*3*4” “1” → $4x^3 + 3x^2 + 2x + 1$
  - “3*-2*4” “1*2” → $\frac{4x^2-2x+3}{2x+1}$
  - “3*0*4” “1*2” “0*2*1” “3*-1”→$\frac{4x^2+3}{2x+1} \cdot \frac{x^2+2x}{-x+3}$

Exercises

- Implement a function that computes the next term in the sequence for Horner’s method given the previous term and the next coefficient
- Implement a function that takes the coefficients of a polynomial and a value x and evaluates the polynomial using Horner’s method
- Implement a function that parses the input string and stores the polynomial coefficients