Additional problems for assignment 2

- **Max of a function**: Define a function max-of-a-func that finds the maximum of a function f in a range based on some arbitrary comparator (a comparison function). It should take the following 3 arguments:
    - range: cons(start, end) where start and end are 3 integers
    - f: which is a function that takes an integer from {start .. end} as argument and returns an integer
    - comparator: which is a comparator function. It can be of various types. For example:
        - comparator(a, b): if (a > b) return #t else return #f
        - comparator(a, b): if (a * a > b * b) return #t else return #f

  How will you design a comparator to find the minimum value of a function in a range?
- **A bit of magic**: Define a function generic-bool(a, b, truth) that applies 'truth' bit-by-bit on a and b and returns the output. Example: generic-bool(5, 6, xor) should return 3 (because 5 ^ 6 == 3). You may assume integers are 32 bits. The function 'truth' takes 2 arguments x and y which are both bits (0 or 1) and returns a bit. Example for the case when truth is xor:
    - xor(0, 0) = 0
    - xor(0, 1) = 1
    - xor(1, 0) = 1
    - xor(1, 1) = 0

  Write various functions like and, or, xor etc. as instances of this.
- **Generic binary search**: Simple idea of binary search in a sorted array can be extended to any monotonic function. Here, we want to apply this idea to find the value x such that f(x) = K. You are given the following:
    - A function f which is monotonic
    - A boolean value that determines if the function is monotonically increasing or decreasing
    - The domain of f is [start, end]. Here start >= 0 and end >= 0 and end > start
    - A number K

  You need to find x such that f(x) = K. If no such x exists, return -1
- **Fixed point iteration:** Implement Newton Raphson method to find the root of a function f. The fixed point iteration for this is

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

  This can be thought of as

$$x_{i+1} = g(x_i)$$

  Take the function g and x_0 as the argument and iterate till the sequence converges.

  Think how will you define convergence

- **Derivative**: The first derivative of a function f(x) can be approximated as

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h}$$

  where h is a 'reasonably small' value. Design a function derivative(f, x) that returns the derivative of a function f at the point x using the above formula. Here, f is a generic function that is to be taken as a parameter. Make necessary assumptions about the value of h. Now extend this to find the n^{th} derivative of f at x.
- **Taylor series**: We know that we can approximate the value of a function '**f**' at a point '**x**' by taylor series expansion. Design a function to estimate the value of f at x given a

$$f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots.$$

  Make assumptions about the number of terms up to which the series has to be computed.