CONCORDIA UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING
SOEN 6011: SOFTWARE ENGINEERING PROCESS: SUMMER 2019

DELIVERABLE 1: OPEN PROBLEMS

STUDENT NAME: MANUSHREE MALLARAJU
STUDENT IDENTIFICATION NUMBER: 40082236

PROBLEM T1-OP5. Source(s): "Give a brief description, not exceeding one page, of your function, including the domain and co-domain of function, and the characteristics that make it unique. To ensure that you have attained sufficient background, Test 1 will have a problem related to your function."

INTRODUCTION

The exponential function $ab^x$is one of the power rules of math, which involves an exponent. This exponent is represented with a variable rather than a constant, and its base is represented with constant value rather than a variable. Let $f(x) = ab^x$ be an exponential function where "b" is its change factor (or a constant), the exponent "x" is the independent variable (or input of the function), the coefficient "a" is called the initial value of the function (or the y-intercept), and "f(x)" represent the dependent variable (or output of the function)

DOMAIN:

The domain is a set of all real numbers,R. where :$b > 0$,$x > 0$

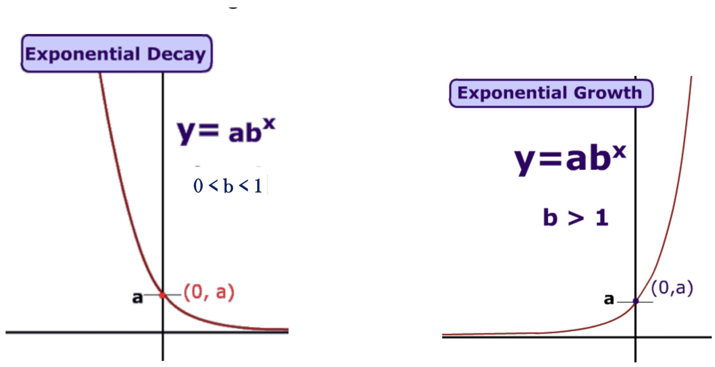
CO DOMAIN

The co-domain is also a set of all real numbers, R.

CHARACTERISTICS

Fig:1 Exponential functions defined by an equation of the form ab^x are called exponential decay functions if the change factor "b" (fixed base value) is $0 < b < 1$, or it is also called exponential growth functions if the change factor is $b > 0$
Fig:2 The y-intercept is (0,a) and it is located at the initial value "a". There is no x-intercept. The domain for an exponential decay function of this form is all real numbers and the range is $y > 0$

GRAPH



(a) Exponential Decay (b) Exponential Growth

Figure 1: Representation of ab^x

PROBLEM 2: Source(s): “Express requirements of your function based on the style given in the ISO/IEC/IEEE 29148 Standard. Associate each requirement with a unique identifier. Make any assumptions explicit. ”

Requirements:

1:

- ID :FR-1
- TYPE :Functional Requirement
- DIFFICULTY :Easy
- VERSION :1.0
- DESCRIPTION : • When a = 0 or b = 0 the function shell simplify to y = f(x) = 0, so all the values should be greater than 1.

2:

- ID :FR-2
- TYPE :Functional Requirement
- DIFFICULTY :Easy
- VERSION :1.0
- DESCRIPTION : When b = 1 the function shell simplify to y = f(x) = a, so all the values of b should be greater than 1.

3:

- ID :FR-3
- TYPE :Functional Requirement
- DIFFICULTY :Easy
- VERSION :1.0
- DESCRIPTION :Expressions with negative bases such as $(-3)^3/2$ or $(-1.4)^2/5$ result in undefined values, the base b in an exponential function must be positive.

EXPLICIT ASSUMPTIONS

1. ’a’ is a constant which can be the combinations of positive or negative numbers.

2. ’b’ is a constant which can be the combinations of positive number.

3. ’x’ is a variable and a real number which values can be in the range between -39 to +39.

PROBLEM 3: Source(s): "Collaboratively brainstorm and mind map with your team members to decide a pseudocode format. The pseudocode format must be identical across the team. Select algorithms for implementing your function and all its subordinate functions, if any. Give technical reasons for selecting each of the algorithms, including their advantages and disadvantages. (This, by reference, means that you must have at least two options to choose from.) Give a brief description of your algorithms and express each of them in pseudocode."

There are several ways to implement the function ab^x.

1. Iteartive Algorithm
2. Recursive Algorithm
3. Divide and Conquer Algorithm

so on..

1. Algorithmic Paradigm: ITERATIVE .

An Iterative algorithm is an approach which executes the steps in number of iterations, using the looping construct. Iterative algorithm uses looping statements such as: for loop, while loop or do-while loop to re-execute the same statements.

Advantages :

- Iterative algorithm is faster to implement, since it uses looping structure.
- Tracing of Iterative algorithm is very easy.
- Iterative algorithm is more efficient with respect to time and space.

Disadvantages :

- Once the condition fails, Iteration terminates the execution of the program. - Iterative solution are high efficient in terms of time and space.

2. Algorithmic Paradigm: RECURSIVE .

A Recursive algorithm is a type of implantation approach in which a function calls by itself during the execution of the program, until the base conditions are met.

Advantages :

- Recursive algorithm reduces unnecessary calling of function.
- Recursive algorithm helps to solve many problems easily.

Disdvantages:

- Understanding of the code in recursive functions is very confusing.
- Since the function calls itself in Recursive, it will be difficult to debug the code.
- In Recursive algorithm, its mandatory to use control statements like if, to return some value from the function.
- Recursive algorithm takes more processor time or memory.

Reason for selecting ITERATIVE ALGORITHM over RECURSIVE ALGORITHM .

* Using Iterative algorithm is it easy to understand the flow of the program, where each step of the iteration is very clear.Whereas with Recursive, since the function calls itself, it is diffult to understand the flow of the program

* Iterative algorithm using looping structure which is comfortable to trace the program, whereas Recursive Algorithm using branching structure.

* If the base conditions are not met, Recursive algorithms have bad performances which leads to infinite call of its own function. Whereas in Iterative, the flow of the program can be easily debugged.

Psuedocode using ITERATIVE ALGORITHM .

Function to calculate power using Newton's method of Nth root

Nth root function

Power function:

```
Input : Base, Exponent
  if Exponent ¿ 0
    while (Int of Exponent ¿ 0)
      result = result * base
      Exponent = Exponent -1
    end while
  else
    while (Int of Exponent ¡ 0)
      result = result * base
      Exponent = Exponent + 1
    end while
    return result
  end if
```

nRoot function:

```
Input: Value, Nthroot
ranVal = 4, precision = 0.0001, x = 1, dx = 2147483647
while(dx ¿ precision)
   x = ((N - 1.0)*ranVal + A/power(ranVal,N-1))/N
   dx = x - ranVal
   if (dx ¡0)
      dx = dx*-1
   precision = x
end while
return x
```

References

[1] Kendall E. Atkinson. nth root algorithm. https://en.wikipedia.org/wiki/Nth_root_algorithm.

[2] BYTES. nth root. https://bytes.com/topic/java/answers/947729-how-find-nth-root-any-positive-number.

[3] CGC. Exponential functions workshop. https://www.cgc.edu/Academics/LearningCenter/Math/Documents.

[4] Barbara Ericson. Integer values. http://ice-web.cc.gatech.edu/ce21/1/static/JavaReview-RU/VariableBasics/minAndMax.html.

[5] Mithun Kumar. nth root number. https://www.geeksforgeeks.org/n-th-root-number/.

[6] MIT. Latex tools. http://kb.mit.edu/confluence/display/istcontrib/How+to+use+Tabs+in+LaTeX.

[7] NNT. Decimal to fraction conversion. http://www.cplusplus.com/forum/general/177760/.

[8] QUORA. Rules of writing the algorithm. https://www.quora.com/What-are-the-rules-in-writing-an-algorithmhttps://www.quora.com/What-are-the-rules-in-writing-an-algorithm.

[9] STACKOVERFLOW. Integer value. https://www.google.com/search?q=integer.max_value&rlz=1C1GCEA_enCA858CA858&oq=Integer.MAX_VALUE&aqs=chrome.0.0l6.175j0j9&sourceid=chrome&ie=UTF-8.

[10] TEX. Pseudo code format. https://tex.stackexchange.com/questions/239697/how-can-i-typeset-an-algorithm-with-columns-as-in-clrs-introduction-to-algorith.