SOEN 6011 Project - Calculator

Suruthi Raju - 40084709

This document is shows basic understanding on the functions that is used in Calculator:

1 Function:

Given functionality for calculation is $a^{b\ x}$

2 Definition:

As per the given function, a and b are real constants and x is a real variable. Real Constants are real number which can be positive, negative or zero, which are fixed values. Whereas Real variables are not fixed values, which may change due to environment. In this project we are calculating the a power of b and b power x.

$b^{\ x}$ = b*....*b which is x times

lets says the above value is $c = b^{\ x}$

$a^{\ c}$ = a*....*a which is c times

3.Domain and Co-Domain of function:

Domain of the power function depend upon value of the power x and b^x

Case 3.1: a is a non-negative integer = then domain is all real numbers

Case 3.2: a is a negative integer = then domain is all real numbers not including zero

Case 3.3: a is a rational number in lowest terms as r/s and s is even = then when a>0 domain is non-negative real numbers, when p<0 domain is positive real numbers

Case 3.4: a is a rational number in lowest terms as r/s and s is odd = then when a>0 domain is all real numbers, when a<0 domain is all real numbers not including zero

Case 3.5: a is an irrational number = then when a>0 domain is all non-negative real numbers, when a<0 domain is all positive real numbers

4.Characteristics:

The properties is known as Exponents power rule

For example: $a^{b\ x}$ is also can be written as $a^{(b\ x)}$
Since a and b are constant lets say 3 and 2 respectively
Where as x are variable it can vary lets say 4

So the above example can be written as $3^{2\ 4}$ or $3^{(2\ 4)}$

so the result is $3^{(2*2*2*2)}$ that is equal to 3^{16} which is 3*....*3 => 16 times

5.Requirements:

Req 5.1: Constant a and b are non-negative integer

Req 5.2: Constant a is a negative integer. Constant b is a negative integer, when x!=0;

Req 5.3: a is a rational number in lowest terms as b=r/s or x=r/s and s is even = then when a>0 domain is non-negative real numbers, when a<0 domain is positive real numbers

Req 5.4: a is a rational number in lowest terms as b=r/s or x=r/s and s is odd = then when a>0 domain is all real numbers, when a<0 domain is all real numbers not including zero

Req 5.5: a is an irrational number = then when a>0 domain is all non-negative real numbers, when a<0 domain is all positive real numbers

6.Assumptions:

Assumption 6.1: The variable value x should not be negative when constant values are zero

Assumption 6.2: The variable value x should not be zero

Assumption 6.3: The constant value a and b should not be zero

7.Pseudocode :

Algorithm 1 Calculate a^{b^x} - for loop

Require: $a > 0 \vee b > 0 \vee x \geq 0$
Ensure: $y = a^{b^x}$

```
y ← 1
z ← 1
a ← input
b ← input
x ← input
while x != 0 do
    z ← z * b
    x ← x − 1
end while
while z != 0 do
    y ← y * a
    z ← z − 1
end while
y = output
```

Technical Reason:
- This algorithm is used for all whole numbers. I am using only integer value for this program/pseudocode.

Advantage:
- Easy to Understand and Calculate.
- Quick in Output.

Disadvantage:
- Can be used only for posivite and whole numbers.
- If number is big, looping takes long time.

Algorithm 2 Calculate a^{b^x} - function recursion

Require: $a > 0 \vee b > 0 \vee x \geq 0$
Ensure: $y = a^{b^x}$
$y \leftarrow 1$
$a \leftarrow input$
$b \leftarrow input$
$x \leftarrow input$
$y \leftarrow FUNCTION(b, x)$
$z \leftarrow FUNCTION(a, y)$
$z = output$
FUNCTION(b,x)
if $x == 0$ **then**
return 1
end if
if $(xmod2) == 0$ **then**
return FUNCTION(b,x/2) * FUNCTION(b,x/2)
else
return b* FUNCTION(b,x/2) * FUNCTION(b,x/2)
end if
FUNCTION(a,y)
if $y == 0$ **then**
return 1
end if
if $(ymod2) == 0$ **then**
return FUNCTION(a,y/2) * FUNCTION(a,y/2)
else
return a* FUNCTION(a,y/2) * FUNCTION(a,y/2)
end if

Technical Reason:
- This algorithm is used for all whole numbers. I am using only integer value for this program/pseudocode.

Advantage:
- Easy to Understand and Calculate.
- Quick in Output.

Disadvantage:
- Can be used only for posivite and whole numbers.
- If number is big, looping takes long time.

Algorithm 3 Calculate a^{b^x} - Math pow() function

Require: $-x \leq a \geq x \vee -x \leq b \geq x \vee x \geq 0 \vee x = infinity$
Ensure: $y = a^{b^x}$
$y \leftarrow 1$
$z \leftarrow 1$
$a \leftarrow input$
$b \leftarrow input$
$x \leftarrow input$
$y = Math.pow(b, x)$
$z = Math.pow(a, y)$
$z = output$

Technical Reason:
- This algorithm is used for all whole numbers including negative integers. I am using this code as part of math library present in Java, so that it can also be used for irrational values.

Advantage:
- can calculate for negative and irrational numbers.
- Quick in Output and easy to understand

Disadvantage:

- Function is inbuilt so changes cannot be made.

REFERENCE:

1. Math-linux.com. (2019). How to write algorithm and pseudocode in Latex ?,- math-linux.com. [online] Available at: https://math-linux.com/latex-26/faq/latex-faq/article/how-to-write-algorithm-and-pseudocode-in-latex-usepackage-algorithm-usepackage-algorithmic

2. Biology.arizona.edu. (2019). BioMath: Power Functions. [online] Available at: http://www.biology.arizona.edu/biomath/tutorials/Power/Powerbasics.html [Accessed 19 Jul. 2019].

3. Oregonstate.edu. (2019). Power Functions. [online] Available at: https://oregonstate.edu/instruct/mth251/cq/FieldGuide/power/lesson.html [Accessed 19 Jul. 2019].