



Assignment 3- Scheme
CSI2120 Programming Paradigm
Winter 2022
Due April 9 11:30 pm via BrightSpace

8%

Question 1: [3 pts]

Write a function `lowest-exponent` that takes as a parameter an integer greater than one (base), and a target natural number n (bound), and returns the lowest positive integer (exponent) for which $\text{base}^{\text{exponent}} \geq \text{bound}$.

For example, `(lowest-exponent 3 27)` returns 3, while `(lowest-exponent 3 28)` returns 4.

Question 2: [2 pts]

A positive integer n is abundant if the sum of all divisors of n are greater than $2n$. As an example, consider the number 24, its divisors are 1, 2, 3, 4, 6, 8, 12, and 24, the sum of which is 60, which is greater than 2×24 , so the number 24 is abundant.

Write a function `find-abundant` that takes as parameters a positive number and produces a list of all abundant numbers no greater than the given one, in order from the largest to the smallest. For example, `(find-abundant 25)` will return `(list 24 20 18 12)`.

Question 3: [3 pts]

Make a program `make-string-list` that takes as parameter n natural number, n , and returns a list of strings. The first string in the list should be the string " n seconds", the second string in the list should be the string " $(n-1)$ seconds" and so on. The second to last string in the list should be "1 second", and the last string in the list should be "Finished".

For example,

`(make-string-list 3)` **should return** `(list "3 seconds" "2 seconds" "1 second" "Finished")`