

CS115 – Lab 11: Trees

Spring 2020

Question 1: That's odd...

Exercise

Create a function (`odd-lengths t`) that consumes a `BST` and returns a `BST` that has the same structure as `t`, but where all values that are strings of odd length are replaced with the string "odd".

```
(define-struct node (key val left right))  
;; A binary search tree (BST) is either  
;; * '() or  
;; * (make-node Nat Any BST BST)...
```

Question 2: Go west, young man...

Exercise

Create a function `leftmost` that consumes a non-empty `SSTree` and returns the leftmost key in the tree.

```
(define-struct snode (key left right))  
;; a SNode is a (make-snode Num SSTree SSTree)  
  
;; a simple search tree (SSTree) is either  
;; * '() or  
;; * a SNode, where keys in left are less than key, and in right greater.
```

Question 3: Adding additions

Exercise

Create a function `count-ops` that consumes a `BinExp` and returns the number of operations in that expression.

```
;; an Operator is (anyof '+ '- '* '/')  
  
define-struct binode (op arg1 arg2)  
;; a binary arithmetic expression internal node (BINode)  
;; is a (make-binode Operator BinExp BinExp)  
  
;; A binary arithmetic expression (BinExp) is either:  
;; a Num or  
;; a BINode
```

Question 4: Four Beautiful Bats, Ah Ah Ah!

Exercise

Create a function (`llt-count value T`) that consumes a `Num` and a `LLT` and counts the number of times that value appears in `T`.

```
(llt-count 7 (list 7 (list 4 (list 7 6 5) 7))) => 3
```

!

Do not use `flatten`.

```
;; a leaf-labelled tree (LLT) is either  
;; a Num or  
;; a non-empty (listof LLT).
```

Question 5: Setting up swap-space

Exercise

Write a function `swap-ops` that consumes an `AExp` and returns a new `AExp` in which the `'+` and `'*` operations have been swapped (i.e. all `'*` become `'+` and all `'+` become `'*`).

```
;; an Operator is (anyof '+ '- '* '/')

(define-struct ainode (op args))
;; an arithmetic expression internal node (AINode)
;; is a (make-ainode Operator (listof AExp))
;; An arithmetic expression (AExp) is either:
;;   a Num or
;;   a AINode
```