

(require spd/tags)

PROBLEMS 1–5:

First consider the signature, purpose, check-expects and st

```
(@signature (listof String) -> Natural)
;; produce the sum of the lengths of the odd length
strings in los
(check-expect (sum-odd-lengths empty) 0)
(check-expect (sum-odd-lengths (list "a")) 1)
(check-expect (sum-odd-lengths (list "ab")) 0)
(check-expect (sum-odd-lengths (list "a" "abc" "ab"
"abcde"))) 9)
```

```
(define (sum-odd-lengths los) 0) ;stub
```

Assignment Project Exam Help

;

You will now write FIVE different definitions for the funct
the signature, purpose, check-expects or stub. Just write
tag and function definition. You will want to use theR odd?

<https://powcoder.com>

Add WeChat powcoder

PROBLEM 1:

Write a definition as a single function based on the (listo
template, with no helper functions. Be sure to include the

;

PROBLEM 2:

Write a definition as a composition of at least 2 helper functions. Be sure to include the @template tag. For each wish, also write a complete wish list entry including signature. You just need the wish list entry for those helper functions that need the complete design for the helper functions.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

;

PROBLEM 3:

Write a definition as a composition of two or more built-in functions. Be sure to include the @template tag.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

PROBLEM 4:

Write a definition as a single tail-recursive function, based on the `(listof String)` template, with a single accumulator and no other arguments. Be sure to include accumulator type, invariant and the @tem

<https://powcoder.com>

Add WeChat powcoder

Assignment Project Exam Help

PROBLEM 5: <https://powcoder.com>

Write a definition as a single function using for-each, with functions. Be sure to include the @template tag.

;

PROBLEM 6:

Design a function that consumes a list of numbers and produces a list of numbers formed by choosing elements of that list as follows.

- the 1st element is included
- the 2nd element of the rest after the above is included
- the 3rd element of the rest after the above is included
- and so on until you run out of elements.

Put another way, it takes a number, then skips 1, takes the number, skips 2, takes the number, skips 3 and so on.

So for example, calling choose as follows

(choose (list 6 1 4 1 3 5 1 2 3 4 1 2 3 4 11 1) 1)

should produce (list 6 4 5 4 11)

Note that the extra spaces in the list above are for illustration purposes only. Your function consumes (listof Number), the no magic spaces. Include all relevant recipe elements, inc purpose, stub, check-expects, @template, accumulator type a

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Assignment Project Exam Help

PROBLEM 7 – 8: <https://powcoder.com>

In these problems you will refactor the definition of `fact` to use built-in abstract functions.

PROBLEM 7:

Write a new definition for `fact` that uses one or more built-in functions.

```
(@signature Natural -> Natural)
;; produce factorial of n
(check-expect (fact 0) 1)
(check-expect (fact 3) 6)
```

```
(@template Natural)
```



```
(define (fact n)
  (cond [(zero? n) 1]
        [else
         (* n (fact (sub1 n)))]))
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

;

PROBLEM 8:

Write a new definition for clip-all that uses one or more b functions.

NOTE: As used below clip means to cut a number down to fit interval.

```
(@signature (listof Number) -> (listof Number))
;; clip every number in list to be within the given [lo,
```

```
hi] interval
(check-expect (clip-all 10 20 (list)) (list))
(check-expect (clip-all 10 20 (list 9 10 11 19 20 21))
              (list 10 10 11 19 20 20))
(check-expect (clip-all 30 50 (list 42 12 39 53 9))
              (list 42 30 39 50 30))

(define (clip-all lo hi lon)
  (cond [(empty? lon) empty]
        [else
         (cons (max lo (min (first lon) hi))
               (clip-all lo hi (rest lon))))]))
```

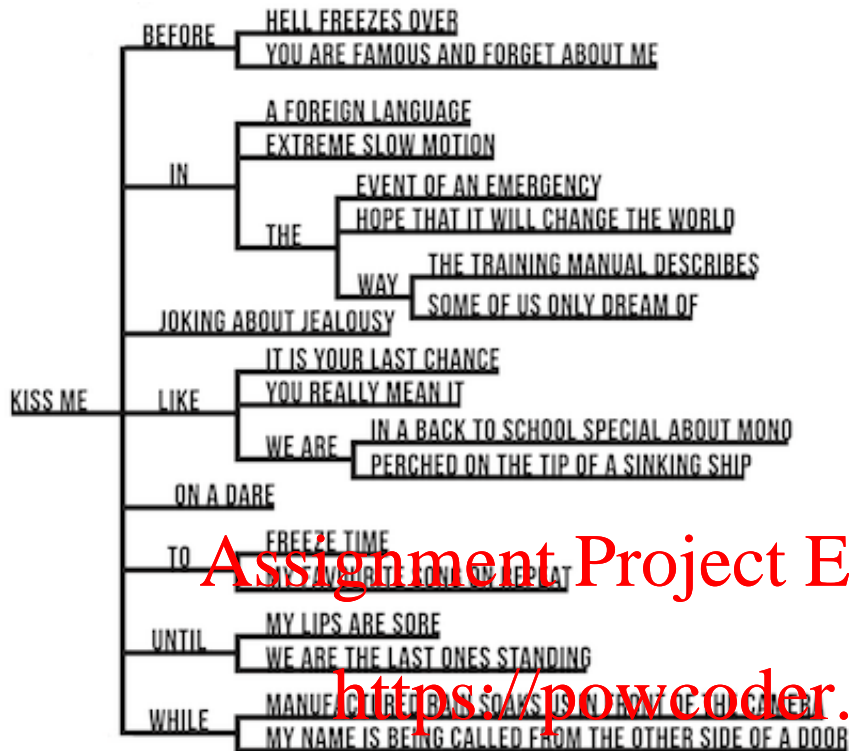
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

PROBLEM 9:

In the mutual reference laboratory you designed data and fu represent and work with sentence trees like these.



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

As you will recall, the idea here is that you can read a nu off of this by following different routes down the tree:

...

KISS ME BEFORE YOU ARE FAMOUS AND FORGET ABOUT ME

KISS ME LIKE IT IS YOUR LAST CHANCE

KISS ME ON A DARE

...

Below are data definitions and example data for sentence tr
Also included is an abstract fold function for STree.

(@htdd STree)

```

(define-struct st (s kids))
;; STree is (make-st String ListOfSTree)
;; interp. (make-st s kids) is a sentence tree with
;;          s as the phrase at the root and
;;          kids as its children

(@htdd ListOfSTree)
;; ListOfSTree is one of:
;; - empty
;; - (cons STree ListOfSTree)
;; interp. A list of sentence trees

(define ST1 (make-st "recipes" empty))
(define ST2 (make-st "instructions for your IKEA
furniture" empty))
(define ST3 (make-st "trust the natural recursion" empty))
(define ST4 (make-st "follow the" (list ST1 ST2)))
(define ST5 (make-st "Make sure you" (list ST3 ST4)))

(@htdf fold-stree)
;; (String Y -> Z) (Z Y -> Y) Y STree -> Z
;; Abstract fold for STree
(check-expect (fold-stree make-st cons empty ST5) ST5)

(define (fold-stree c1 c2 b1 st)
  (local [(define (fn-for-stree st)
            (c1 (st-s st)
                (fn-for-forest (st-kids st))))

          (define (fn-for-forest lost)
            (cond [(empty? lost) b1]
                  [else
                   (c2 (fn-for-stree (first lost))
                       (fn-for-forest (rest lost))))])])
    (fn-for-stree st)))

```

Please design a function that consumes an STree and produce the sentences that can be generated starting at the root of Each sentence should be a list of the strings that form the So calling your function with ST1 would produce (list (list because ST1 just describes a single sentence, and that sing is formed from just one contribution "recipes". Calling with a list of two lists, each with two strings.

You must provide signature, purpose, stub, check-expects an Your function must call fold-stree, no credit will be given

Work systematically and pay particular attention to the example of the recipe. While this problem is not particularly difficult it is a little more complicated than it seems at first.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Problem 10:

READ THIS BOX TWICE CAREFULLY BEFORE BEGINNING TO WORK ON T

Below is a simple solver for square mazes. It has been PAR from 2 way mazes into a solver for 4 way mazes. What has b is to update the next-ps function to support moves in all f But nothing has been done to prevent going in cycles in the

You must complete the conversion and add the following new

Instead of searching for just one path, the function should without cycles. If there is just one, produce it. If there one, produce the shortest. If there are none produce fals mazes are small, you do not have to use tail recursion for

So (solve M1) should produce

```
(list (make-pos 0 0) (make-pos 0 1) (make-pos 1 1) (make-po
      (make-pos 1 3) (make-pos 1 4) (make-pos 2 4) (make-po
      (make-pos 4 4))
```

(solve M7) produces the path down the left edge and across edge of the maze since that is the shortest.

YOU MUST COMPLETELY CONVERT THE SOLVE FUNCTION TO THIS NEW

BUT, don't panic!! The changes required are quite few, and understand them, not that complex. It will really pay to think about templates and think about combination positions

The check expects will be cumbersome. To save you time you write out the result as a list of make-pos. You can instead M1 to M7 constants to show what calling solve would produce

Be sure to edit the signature, purpose, and @template tag a

Also edit the actual function definition. Our solution makes changes to solve/p and solve/lop and adds a single new help

Work systematically – draw pictures, think carefully about the @template, then be careful about doing that in the code

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

```
;; Solve simple square mazes
```

```
;; Data definitions:
```

```
(@htdd Maze)
```

```
;; Maze is (listof Boolean)
```

```
;; interp. a square maze
```

```
;;          each side length is (sqrt (length <maze>))
```

```
;;          true (aka #t) means open, can move through this
```

```
;;          false (aka #f) means a wall, cannot move into
```

```
or through a wall
```

```
;;
```

```
(define O #t) ;Open
```

```
(define W #f) ;Wall
```

```
(define M1
  (list 0 W W W W
        0 0 W 0 0
        W 0 W W W
        0 0 W W W
        0 0 0 0 0))
```

```
(define M2
  (list 0 0 0 0 0
        0 W W W 0
        0 W W W 0
        0 W W W 0
        0 W W W 0))
```

```
(define M3
  (list 0 0 0 0 0
        0 W W W W
        0 W W W W
        0 W W W W
        0 0 0 0 0))
```

```
(define M4
  (list 0 0 0 0 0
        0 W W W 0
        0 W 0 0 0
        0 W 0 W W
        W W 0 0 0))
```

```
(define M5
  (list 0 0 0 0 0
        0 W 0 W 0
        0 0 0 0 0
        0 W 0 W 0
        W 0 0 W W))
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder


```
(define M6
  (list 0 0 0 0 0
        0 W 0 W 0
        0 0 0 0 0
        0 W 0 W 0
        W 0 0 0 0))
```

```
(define M7
  (list 0 0 0 0 0 0 0
        0 W W W W W 0
        0 W 0 0 0 W 0
        0 W 0 W 0 0 0
        0 W 0 W W W W
        0 W 0 W W 0 0
        0 0 0 0 0 0 0))
```

Assignment Project Exam Help

```
(@htdd Pos)
```

```
(define-struct pos (x y))
```

```
;; Pos is (make-pos Integer Integer)
```

```
;; interp. an x, y position in the maze.
```

```
;; 0, 0 is upper left.
```

```
;; the SIZE of a maze is (sqrt (length m))
```

```
;; a position is only valid for a given maze if:
```

```
;; - (<= 0 x (sub1 SIZE))
```

```
;; - (<= 0 y (sub1 SIZE))
```

```
;; - there is a true in the given cell
```

```
;; ;in a 5x5 maze:
```

```
(define P0 (make-pos 0 0)) ;upper left
```

```
(define P1 (make-pos 4 0)) ;upper right
```

```
(define P2 (make-pos 0 4)) ;lower left
```

```
(define P3 (make-pos 4 4)) ;lower right
```

```
;; Functions
```

```
(@signature Maze -> Boolean)
```

```
;; produce true if maze is solvable, false otherwise
```

```
;; assume maze has a true at least in the upper left
```

```
(check-expect (solve M1) #t)
(check-expect (solve M2) #t)
(check-expect (solve M3) #t)
(check-expect (solve M4) #t)
(check-expect (solve M5) #f)
(check-expect (solve M6) #t)
(check-expect (solve M7) #t)
```

```
(@template encapsulated backtracking genrec arb-tree)
```

```
(define (solve m)
  (local [(define S (sqrt (length m)))]
```

```
    (define (solve/p p)
```

```
      (cond [(solved? p) true]
```

Assignment Project Exam Help

```
      [else
```

<https://powcoder.com>

```
        (solve/lop (next-ps p))]]))
```

Add WeChat powcoder

```
(define (solve/lop lop)
```

```
  (cond [(empty? lop) false]
```

```
  [else
```

```
    (local [(define try (solve/p (first
lop))))]
```

```
    (if (not (false? try))
```

```
      try
```

```
      (solve/lop (rest lop))))))])
```

```
<this space intentionally left blank for you>
<rest of local is on next page          >
```

```
;; Pos -> Boolean
;; produce true if p is bottom right
;; (@template Pos)
(define (solved? p)
  (and (=< (pos-x p) (sub1 S))
        (= (pos-y p) (sub1 S))))

;; Pos -> (listof Pos)
;; produce list Up, down, left, right, but only
when they are valid
;; (@template use-abstract-fn)
(define (next-ps p)
  (local [(define x (pos-x p))
          (define y (pos-y p))]
    (filter (lambda (p1)
              (and (<= 0 (pos-x p1) (sub1 S))
                   (<= 0 (pos-y p1) (sub1 S))
                   (maze-ref p1))))
    (list (make-pos (add1 x) y)      ;R
          (make-pos (sub1 x) y)      ;L
          (make-pos x (sub1 y))      ;U
```

```
(make-pos x (add1 y)))))) ;D
```

```
;; Maze Pos -> Boolean  
;; produce value of the maze cell at position p  
;; (@template Pos)  
(define (maze-ref p)  
  (list-ref m  
    (+ (* (pos-y p) S) (pos-x p))))]  
  
(solve/p (make-pos 0 0)))
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder