Question 2-5

Bookmark this page

Here is our data definition for a list of strings:

```
;; ListOfString is one of:
;; - empty
;; - (cons String ListOfString)
;; interp. a list of strings

(define LOS1 empty)
(define LOS2 (cons "a" empty))
(define LOS3 (cons "a" (cons "b" empty)))

#;
(define (fn-for-los los)
  (cond [(empty? los)(...)]
        [else
         (... (first los)
              (fn-for-los (rest los)))]))
```

Question 2

1/1 point (graded)

Here is the signature and purpose for `arrange-strings`:

```
;; ListOfStrings -> Image
;; layout strings vertically in alphabetical order
```

Which of the following is a correct test for `arrange-strings`?

○
```
(check-expect (arrange-strings (cons "Sally" (cons "Apple" empty))) BLANK)
```

○
```
(check-expect (arrange-strings (cons "Sally" (cons "Apple" empty)))
              (above/align "left"
                           (text "Sally" TEXT-SIZE TEXT-COLOR)
                           (text "Apple" TEXT-SIZE TEXT-COLOR)
                           BLANK))
```

●
```
(check-expect (arrange-strings (cons "Sally" (cons "Apple" empty)))
              (above/align "left"
                           (text "Apple" TEXT-SIZE TEXT-COLOR)
                           (text "Sally" TEXT-SIZE TEXT-COLOR)
                           BLANK))
```

✓

Explanation
First we ned to sort the list alphabetically, and then lay the elements out vertically using the `above/align` and `text` functions.

Submit Show Answer

Answers are displayed within the problem

Question 3

1/1 point (graded)

Do we need to add the following base case test to the `arrange-strings` function design?

```
(check-expect (arrange-strings empty) BLANK)
```

○ yes

● no

✓

Explanation
Because `arrange-strings` is a function composition, the tests only need to ensure that `arrange-strings` is composing the functions properly, so no base case test is needed.

Submit

Show Answer

Answers are displayed within the problem

Question 4

1/1 point (graded)

Consider the following `check-expect`:

```
(check-expect (arrange-strings (cons "Apple" (cons "Sally" empty))) BLANK)
              (above/align "left"
                           (text "Apple" TEXT-SIZE TEXT-COLOR)
                           (text "Sally" TEXT-SIZE TEXT-COLOR)
                           BLANK))
```

Is it good enough to have only this `check-expect` for `arrange-strings`?

○ yes

● no

✓

Explanation
The list is already sorted alphabetically, so this tests the rendering part, but not the sorting.

Submit

Show Answer

Answers are displayed within the problem

Question 5

1/1 point (graded)

Since `arrange-strings` needs to perform two tasks, we need to use a function composition. Assume we wished for two functions, `sort-strings` which sorts a list of strings, and `layout-strings` that arranges the strings into a vertical line.

Which of the following is the correct function definition for arrange strings?

●

```
(define (arrange-strings los)
  (layout-strings (sort-strings los)))
```

○

```
(define (arrange-strings los)
  (sort-strings (layout-strings los)))
```

✓

Explanation
The first task is to sort the strings, and the second is to lay them out. Remember the BSL evaluation rules. Expressions like this evaluate inside to outside. So the sorting evaluates before the laying out, because the call to the sorting function is an operand to the call to the laying out function.

Submit

Show Answer

Answers are displayed within the problem

< Previous | Next >