Question 1

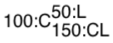
Question 1

1/1 point (graded)

You're still not comfortable with the idea that trees are layed out from above to below in computer science, instead you would prefer to have your trees layed out left to right. So the root is at the left, it's children are to it's right etc. For example, with your new version the result of evaluating:

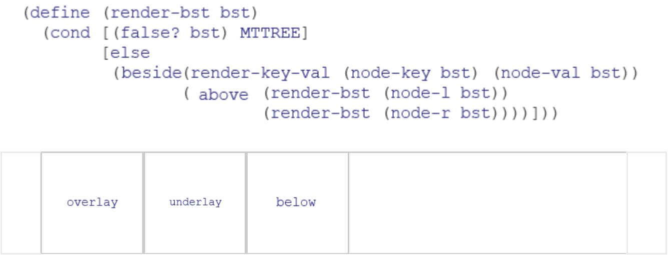
```
(render-bst (make-node 100 "C"
                       (make-node  50  "L" false false)
                       (make-node 150 "CL" false false)))
```

would look something like:



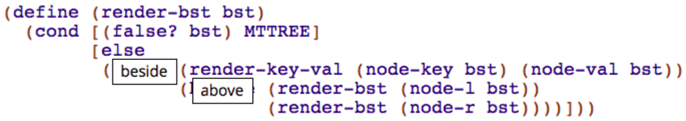
Again, that would look a lot better with a little extra whitespace in it, but please ignore that for now.

By remembering the role that combination positions play in templates, you realize that the new function definition should be very close to the old function definition. Edit the current render function definition so that it has the new left-to-right behavior by dragging one or more function names from the scrollbar below, onto the function definition where you want to replace one function name with another.



```
(define (render-bst bst)
  (cond [(false? bst) MTTREE]
        [else
         (beside(render-key-val (node-key bst) (node-val bst))
                ( above (render-bst (node-l bst))
                        (render-bst (node-r bst))))]))
```

✓



```
(define (render-bst bst)
  (cond [(false? bst) MTTREE]
        [else
         (beside (render-key-val (node-key bst) (node-val bst))
                 (above (render-bst (node-l bst))
                        (render-bst (node-r bst))))]))
```

Submit

Answers are displayed within the problem