* 1. A tail-recursive function is one whose recursive function call is its final operation. This function fails to meet that definition by cons’ing *hd* onto the result of the recursive call, meaning that the recursive call itself is only the second-last operation



* 1. If for example, l1 has many elements, the standard library implementation might cause a stack overflow error. The tail-recursive implementation however can use TRO to discard unnecessary stack frame data and will not cause an error.
  2. CPS-converted functions can lead to messy and hard-to-read code, as the caller of the function must remember to pass a continuation function (often the identity).



1. The type does not change







2017 P3 Q4

