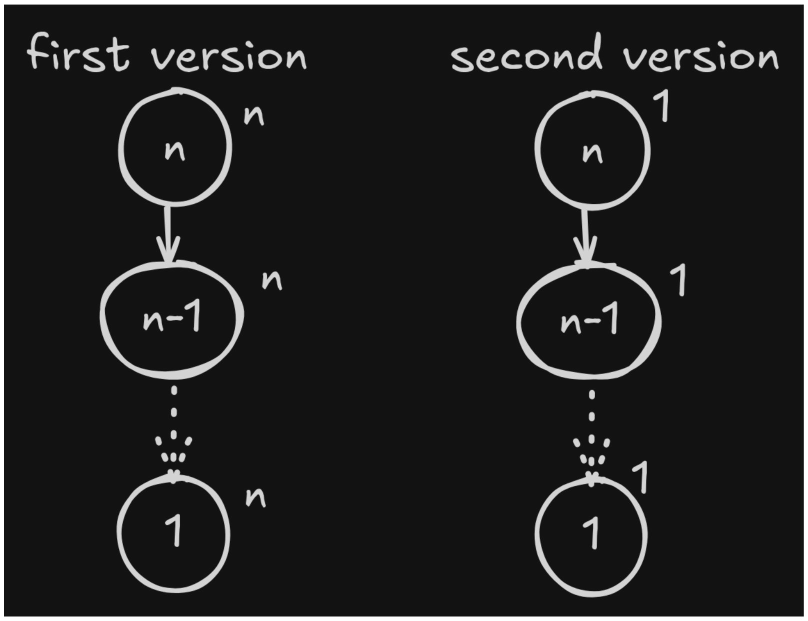
1)

base case: return last element if its got length of 1

recusrion: create sublist of list[1:] and add to index 0, keep shrinking list till base case is reached

2)



- first version
  - each recursive call makes a slice of list costing O(n-i) where i increases by 1 each for each recursive call making total time into $n(n-1)$ which gives $O(n^2)$ (quadrati ctime)
- second version
  - each call performs only constant time operation sthere fore the total number of calls is n each taking constant time making the total $O(n^2)$ (linear time)

3)

second version is much faster since it takes less time per call