```
# Complete the stepPerms function below.
def stepPerms(n):
  # when we're looking at how many steps we can take, we

  # we can use memoization!
      # basically we check if this combination is in our
      # calculate and add it. If it is, then we just ret

  # we could either loop one step, two steps, or three s
  memo = {1: 1, 2: 2, 3: 4}
  #
  return stepPermsRecursive(n, memo)
  # maybe we create a helper function
  # base case: is it present in the memo, if not then we

def stepPermsRecursive(n, memo):
  # base case - check if our object already exists in me
  # if n in memo:
  #   return memo[n]

  if n not in memo:
    memo[n] = stepPermsRecursive(n-1,memo) + stepPermsRe
    # call our recursive function, and we're going to ca
    # call it for taking 1 step, 2 steps, and 3 steps, t
  # store in our memo if our object does not exist

  return memo[n]

```

▶ Run