C-4.10

Simply use a temporary node to walk to the end of list L. Then, make the last element of L point to the first element of M as its "next" node. The running time of this method is $O(n)$.

Algorithm Concatenate():

Create a new node v
$v = L$.getHead()
while $v != \text{null}$ do
  $v = V$.getNext()
v.setNext(M.getHead())
$L' = L$