Problem 4

a) in1 = 1,2,3,4 in2 = 5,6

call: llrec([1,2,3,4],[5,6])

in1: 1,2,3,4 in1->next: 2,3,4 in2: 5,6

in1->next = llrec(in2, in1->next)

in1->next = llrec([5,6],[2,3,4])

return in1 = return [1, in1->next]

call: llrec([5,6],[2,3,4])

in1: 5,6 in1->next: 6 in2: 2,3,4

in1->next = llrec(in2, in1->next)

in1->next = llrec([2,3,4],[6])

return in1 = return [5, in1->next]

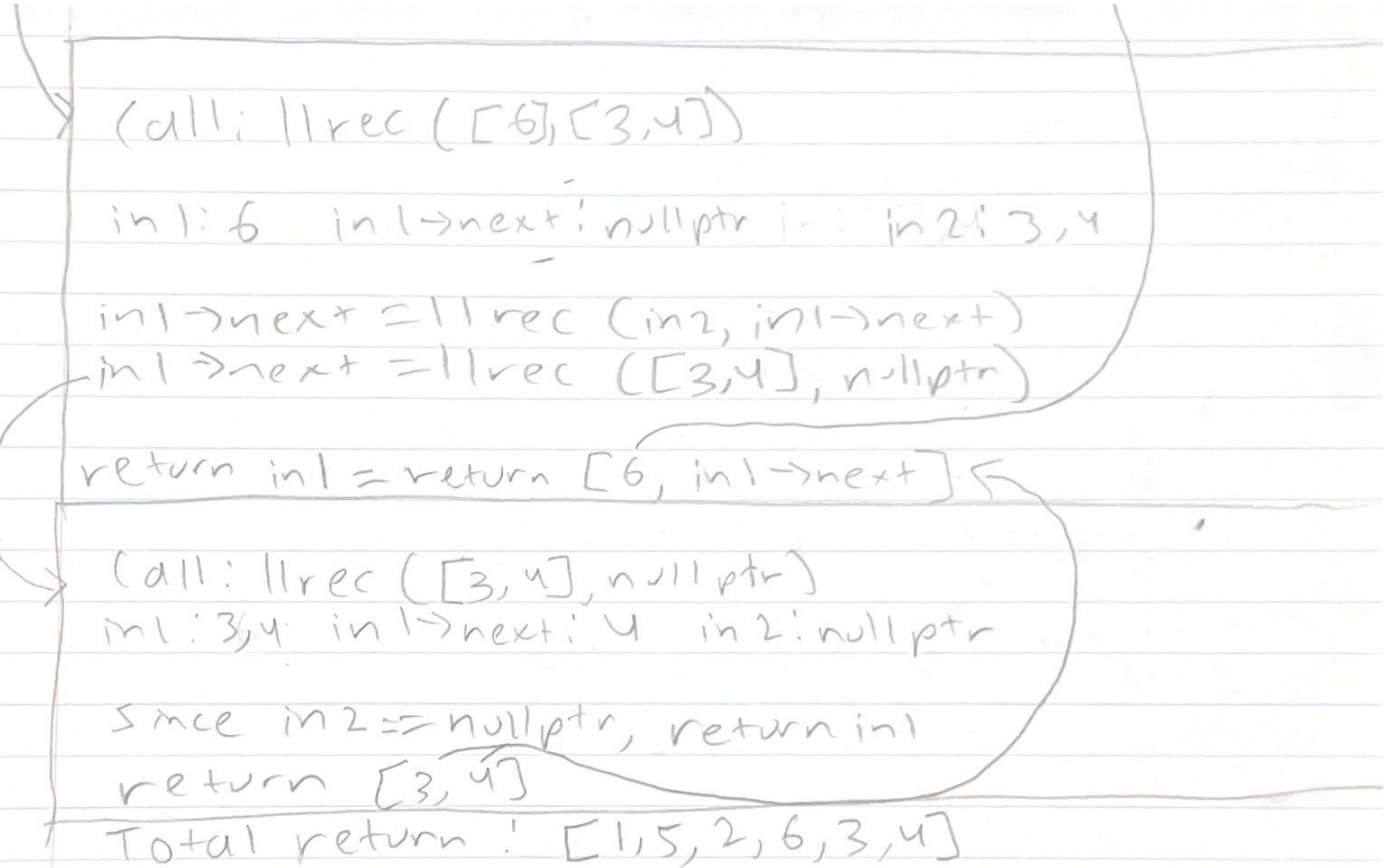
call: llrec([2,3,4],[6])

in1: 2,3,4 in1->next: 3,4 in2: 6

in1->next = llrec(in2, in1->next)

in1->next = llrec([6],[3,4])

return in1 = return [2, in1->next]



Inside the boxes there are the values of each node, where the call came from, and what the call returns. Since the function recursively adds to the end of a linked list on every call, the final linked list that the function returns is [1,5,2,6,3,4].

Problem 4

b) in1 = nullptr in2 = 2

llrec (nullptr, [2])

Since in1 == nullptr, the function will return in2, which is the linked list [2].