**CSE220 Lab quiz04 (Recursion)**

**SET A:**

Function findLength(node):

If node is NULL:

Return 0

Return 1 + findLength(node.next)

Function getMiddleValue(node, index, targetIndex):

If index == targetIndex:

Return node.elem

Return getMiddleValue(node.next, index + 1, targetIndex)

Function multiplyAllWith(node, value):

If node is NULL:

Return NULL

node.elem = node.elem \* value

node.next = multiplyAllWith(node.next, value)

Return node

Function multiply\_with\_middle\_setB(head):

length = findLength(head)

# For even, pick the second middle: index = length // 2

midIndex = length // 2

middleValue = getMiddleValue(head, 0, midIndex)

head = multiplyAllWith(head, middleValue)

Return head

**RUBRIC:**

1. Finding the middle value using recursion - 5
2. Multiply all the values using recursion - 5
3. Finding the length of the linked list recursively - 2.5
4. Doing all the tasks without using any loops and only using recursion - 2.5

**SET B:**

Function findLength(node):

If node is NULL:

Return 0

Return 1 + findLength(node.next)

Function getMiddleValue(node, index, targetIndex):

If index == targetIndex:

Return node.elem

Return getMiddleValue(node.next, index + 1, targetIndex)

Function divideAllWith(node, value):

If node is NULL:

Return NULL

node.elem = node.elem // value

node.next = divideAllWith(node.next, value)

Return node

Function divide\_middle(head):

length = findLength(head)

# For even, pick the second middle: index = length // 2

midIndex = length // 2

middleValue = getMiddleValue(head, 0, midIndex)

head = divideAllWith(head, middleValue)

Return head

**RUBRIC:**

1. Finding the middle value using recursion - 5
2. Divide all the values using recursion - 5
3. Finding the length of the linked list recursively - 2.5
4. Doing all the tasks without using any loops and only using recursion - 2.5