Delete a Node ★

274 more points to get your next star!

Rank: 1062200 | Points: 201/475

Problem Solving ★★★

Your Delete a Node submission got 5.00 points. Share Post

You are now 274 points away from the 4th star for your problem solving badge.

Try the next challenge | Try a Random Challenge

Problem Submissions Leaderboard Editorial

This challenge is part of a tutorial track by MyCodeSchool and is accompanied by a video lesson.

Delete the node at a given position in a linked list and return a reference to the head node. The head is at position 0. The list may be empty after you delete the node. In that case, return a null value.

Example

$llist = 0 \rightarrow 1 \rightarrow 2 \rightarrow 3$

$position = 2$

After removing the node at position 2, $llist' = 0 \rightarrow 1 \rightarrow 3$.

Function Description

Complete the deleteNode function in the editor below.

deleteNode has the following parameters:

- SinglyLinkedListNode pointer llist: a reference to the head node in the list
- int position: the position of the node to remove

Returns

- SinglyLinkedListNode pointer: a reference to the head of the modified list

Input Format

The first line of input contains an integer n, the number of elements in the linked list.

Each of the next n lines contains an integer, the node data values in order.

The last line contains an integer, $position$, the position of the node to delete.

Constraints

- $1 \le n \le 1000$
- $1 \le list[i] \le 1000$, where $list[i]$ is the i^{th} element of the linked list.

Sample Input

```
8
20
6
2
19
7
4
15
9
3
```

Sample Output

```
20 6 2 7 4 15 9
```

Explanation

The original list is $20 \rightarrow 6 \rightarrow 2 \rightarrow 19 \rightarrow 7 \rightarrow 4 \rightarrow 15 \rightarrow 9$. After deleting the node at position 3, the list is $20 \rightarrow 6 \rightarrow 2 \rightarrow 7 \rightarrow 4 \rightarrow 15 \rightarrow 9$.

Change Theme Language Python 3

```
# The function accepts following parameters:
#  1. INTEGER_SINGLY_LINKED_LIST llist
#  2. INTEGER position
#

#
# For your reference:
#
# SinglyLinkedListNode:
#     int data
#     SinglyLinkedListNode next
#
#

def deleteNode(llist, position):
    # Write your code here
    cur = llist
    pre = None
    for _ in range(position):
        pre = cur
        cur = cur.next
    # delete the head node
    if cur == llist:
        # only one node
        if llist.next is None:
            return None
        else:
            # there are other node left
            return llist.next
    # delete inner node
    pre.next = cur.next
    return llist

if   name   == '  main  ':⋯
```

EMACS Line: 44 Col: 1

Upload Code as File ☐ Test against custom input Run Code Submit Code

You have earned 5.00 points!

You are now 274 points away from the 4th star for your problem solving badge.

0% 201/475

Problem Solving ★★★

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

✓ **Test case 0**

✓ Test case 1

✓ Test case 2 🔒

✓ Test case 3 🔒

✓ Test case 4 🔒

✓ Test case 5 🔒

✓ Test case 6 🔒

Compiler Message

```
Success
```

Input (stdin) Download

```
8
20
6
2
19
7
4
15
9
```

Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy