Your Delete a Node submission got 5.00 points. Share Tweet

Try the Next Challenge|Try a Random Challenge

Delete a Node

by harsha_s

Problem | Submissions | Leaderboard | Discussions | Editorial

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

You're given the pointer to the head node of a linked list and the position of a node to delete. Delete the node at the given position and return the head node. A position of 0 indicates head, a position of 1 indicates one node away from the head and so on. The list may become empty after you delete the node.

Input Format

You have to complete the `Node* Delete(Node* head, int position)` method which takes two arguments - the head of the linked list and the position of the node to delete. You should NOT read any input from stdin/console. `position` will always be at least 0 and less than the number of the elements in the list.

Output Format

Delete the node at the given position and `return` the head of the updated linked list. Do NOT print anything to stdout/console.

Sample Input

1 --> 2 --> 3 --> NULL, position = 0
1 --> NULL , position = 0

Sample Output

```
2 --> 3 --> NULL
NULL
```

Explanation

1. 0th position is removed, 1 is deleted from the list.
2. Again 0th position is deleted and we are left with empty list.

Video lesson

Submissions:55901

Max Score:5

Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable) C++14

```
/*
  Delete Node at a given position in a linked list
  Node is defined as
  struct Node
```

```
  {
     int data;
     struct Node *next;
  }
*/
Node* Delete(Node *head, int position)
{
  struct Node *current;
    int pos=0;
    current = head;

    if(position == 0)
        {
            head = head->next;

        }
    else
    {while(pos<position-1)
            {
                current = current->next;
                pos++;
            }
        current->next = current->next->next;

    }
    return head;
}

```

Line: 19 Col: 13

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

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5
- ✔ Test Case #6
- ✔ Test Case #7
- ✔ Test Case #8
- ✔ Test Case #9

You've earned 5.00 points! Next Challenge