

# GeeksQuiz

Computer science mock tests for geeks

- [Home](#)
- [Latest Questions](#)
- [Articles](#)
- [C/C++ Programs](#)
- [Contribute](#)
- [Books](#)

[Subscribe](#)

## Linked List | Set 3 (Deleting a node)

May 24, 2014

We have discussed [Linked List Introduction](#) and [Linked List Insertion](#) in previous posts on singly linked list.

Let us formulate the problem statement to understand the deletion process. *Given a 'key', delete the first occurrence of this key in linked list.*

To delete a node from linked list, we need to do following steps.

- 1) Find previous node of the node to be deleted.
- 2) Changed next of previous node.
- 3) Free memory for the node to be deleted.

Since every node of linked list is dynamically allocated using malloc() in C, we need to call [free\(\)](#) for freeing memory allocated for the node to be deleted.

```
// A complete working C program to demonstrate deletion in singly
// linked list
#include <stdio.h>
#include <stdlib.h>

// A linked list node
struct node
{
    int data;
    struct node *next;
};

/* Given a reference (pointer to pointer) to the head of a list
   and an int, inserts a new node on the front of the list. */
void push(struct node** head_ref, int new_data)
{
```

```

    struct node* new_node = (struct node*) malloc(sizeof(struct node));
    new_node->data = new_data;
    new_node->next = (*head_ref);
    (*head_ref) = new_node;
}

/* Given a reference (pointer to pointer) to the head of a list
   and a key, deletes the first occurrence of key in linked list */
void deleteNode(struct node **head_ref, int key)
{
    // Store head node
    struct node* temp = *head_ref, *prev;

    // If head node itself holds the key to be deleted
    if (temp != NULL && temp->data == key)
    {
        *head_ref = temp->next;    // Changed head
        free(temp);                // free old head
        return;
    }

    // Search for the key to be deleted, keep track of the
    // previous node as we need to change 'prev->next'
    while (temp != NULL && temp->data != key)
    {
        prev = temp;
        temp = temp->next;
    }

    // If key was not present in linked list
    if (temp == NULL) return;

    // Unlink the node from linked list
    prev->next = temp->next;

    free(temp);    // Free memory
}

// This function prints contents of linked list starting from
// the given node
void printList(struct node *node)
{
    while (node != NULL)
    {
        printf(" %d ", node->data);
        node = node->next;
    }
}

/* Driver program to test above functions*/
int main()
{
    /* Start with the empty list */

```

```

struct node* head = NULL;

push(&head, 7);
push(&head, 1);
push(&head, 3);
push(&head, 2);

puts("Created Linked List: ");
printList(head);
deleteNode(&head, 1);
puts("\nLinked List after Deletion of 1: ");
printList(head);
return 0;
}

```

Output:

```

Created Linked List:
2 3 1 7
Linked List after Deletion of 1:
2 3 7

```

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

## Related Questions:

- [Binary Heap](#)
- [Delete all occurrences of a given key in a linked list](#)
- [How to create mergable stack?](#)
- [Deque | Set 1 \(Introduction and Applications\)](#)
- [A data structure for n elements and O\(1\) operations](#)
- [Convert left-right representation of a bianry tree to down-right](#)
- [Print level order traversal line by line](#)
- [C Program for Red Black Tree Insertion](#)

Like { 4

Tweet { 1

g+1 { 1

- 
- 

## • Categories

- [Articles](#) (128)
  - [Algorithms](#) (24)
  - [C](#) (17)

- [C++](#) (17)
  - [Data Structures](#) (29)
  - [DBMS](#) (1)
  - [Interview Experiences](#) (6)
  - [Java](#) (2)
  - [Operating Systems](#) (1)
  - [Puzzle](#) (15)
  - [Searching and Sorting](#) (10)
  - [Programs](#) (35)
  - [Quizzes](#) (2,107)
    - [Aptitude](#) (2)
    - [Computer Science Quizzes](#) (2,104)
      - [Algorithms](#) (147)
      - [C](#) (208)
      - [C++](#) (129)
      - [Computer Organization and Architecture](#) (1)
      - [Data Structures](#) (132)
      - [DBMS](#) (2)
      - [GATE](#) (1,406)
      - [Java](#) (51)
      - [Operating Systems](#) (28)
    - [Web technologies](#) (1)
- 

## • Recent Discussions

- [Zsw-seu](#)

I agree your answer,it should be n-1

[Bubble Sort](#) · [2 hours ago](#)

- [vedraiyani](#)

I have run it in codeblock giving output 4

[C | Operators | Question 5](#) · [9 hours ago](#)

- [dileep kumar Grandhi](#)

please explain how "s" has come in output

[C | Operators | Question 12](#) · [11 hours ago](#)

- [ASBGeek](#)

Can someone explain how the assigement...

[GATE | GATE-CS-2014-\(Set-2\) | Question 21](#) · [11 hours ago](#)

- [Zsw-seu](#)

I think there is a problem with binarySearch....

[Binary Insertion Sort](#) · [20 hours ago](#)

- Kuch\_Bhi

Please UPDATE the answer. option A is correct.

[GATE | GATE-CS-2003 | Question 57](#) · [22 hours ago](#)

•

Valid [XHTML Strict 1.0](#)

Powered by [WordPress](#) & [MooTools](#) | MiniMoo 1.3.4