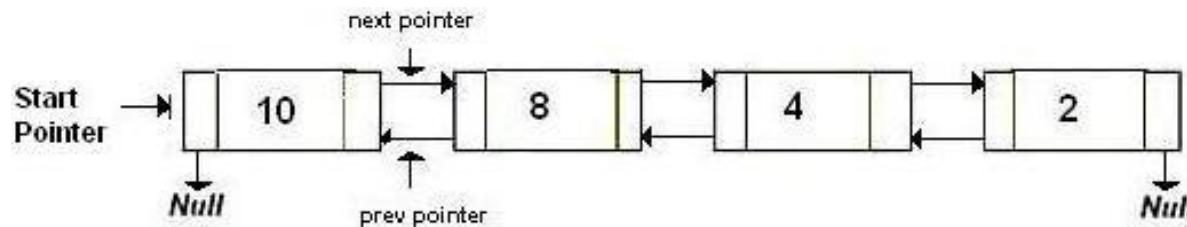


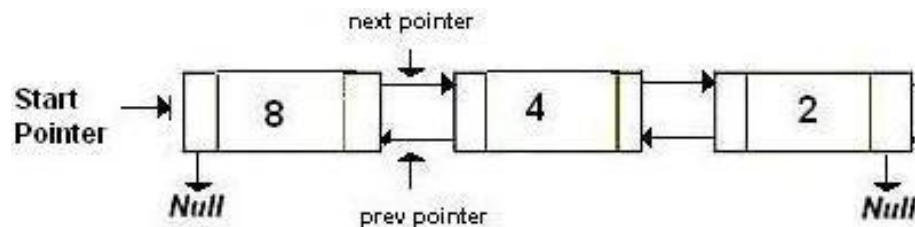
Delete a node in a Doubly Linked List

Write a function to delete a given node in a doubly linked list.

(a) Original Doubly Linked List



(a) After deletion of head node



(a) After deletion of middle node

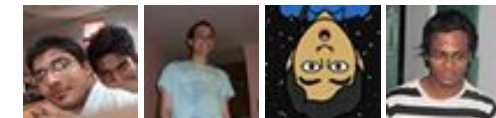
Google™ Custom Search



GeeksforGeeks



53,528 people like GeeksforGeeks.



Facebook

Interview Experiences

Advanced Data Structures

Dynamic Programming

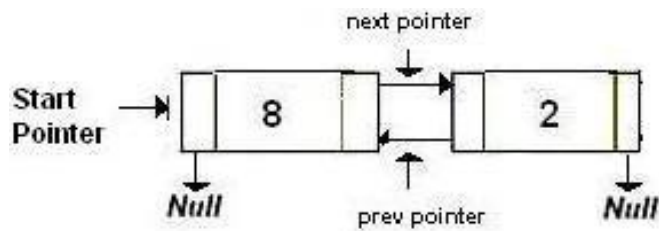
Greedy Algorithms

Backtracking

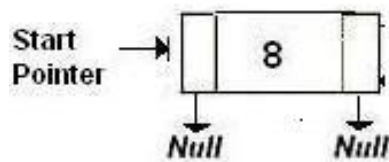
Pattern Searching

Divide & Conquer

Mathematical Algorithms



(a) After deletion of last node



Algorithm

Let the node to be deleted is *del*.

- 1) If node to be deleted is head node, then change the head pointer to next current head.
- 2) Set *next* of previous to *del*, if previous to *del* exists.
- 3) Set *prev* of next to *del*, if next to *del* exists.

```

#include <stdio.h>
#include <stdlib.h>

/* a node of the doubly linked list */
struct node
{
    int data;
    struct node *next;
    struct node *prev;
};

/* Function to delete a node in a Doubly Linked List.
   head_ref --> pointer to head node pointer.
   del --> pointer to node to be deleted. */
void deleteNode(struct node **head_ref, struct node *del)
{
    /* base case */
    if(*head_ref == NULL || del == NULL)
  
```

Recursion

Geometric Algorithms

Build Applications Without Coding

Under the Sea Productions

The ACME FOOD COMPANY

Free Download!

IRON SPEED

Popular Posts

All permutations of a given string

Memory Layout of C Programs

Understanding "extern" keyword in C

Median of two sorted arrays

Tree traversal without recursion and without stack!

Structure Member Alignment, Padding and Data Packing

Intersection point of two Linked Lists

Lowest Common Ancestor in a BST.

Check if a binary tree is BST or not

Sorted Linked List to Balanced BST

```

return;

/* If node to be deleted is head node */
if(*head_ref == del)
    *head_ref = del->next;

/* Change next only if node to be deleted is NOT the last node */
if(del->next != NULL)
    del->next->prev = del->prev;

/* Change prev only if node to be deleted is NOT the first node */
if(del->prev != NULL)
    del->prev->next = del->next;

/* Finally, free the memory occupied by del*/
free(del);
return;
}

/* UTILITY FUNCTIONS */
/* Function to insert a node at the beginning of the Doubly Linked List */
void push(struct node** head_ref, int new_data)
{
    /* allocate node */
    struct node* new_node =
        (struct node*) malloc(sizeof(struct node));

    /* put in the data */
    new_node->data = new_data;

    /* since we are adding at the beginning,
       prev is always NULL */
    new_node->prev = NULL;

    /* link the old list off the new node */
    new_node->next = (*head_ref);

    /* change prev of head node to new node */
    if((*head_ref) != NULL)
        (*head_ref)->prev = new_node;

    /* move the head to point to the new node */
    (*head_ref) = new_node;
}

/* Function to print nodes in a given doubly linked list
   This function is same as printList() of singly linked list */

```

Market research
that's fast and
accurate.

Get \$75 off

 Google consumer surveys

```

void printList(struct node *node)
{
    while (node!=NULL)
    {
        printf("%d ", node->data);
        node = node->next;
    }
}

/* Drier program to test above functions*/
int main()
{
    /* Start with the empty list */
    struct node* head = NULL;

    /* Let us create the doubly linked list 10<->8<->4<->2 */
    push(&head, 2);
    push(&head, 4);
    push(&head, 8);
    push(&head, 10);

    printf("\n Original Linked list ");
    printList(head);

    /* delete nodes from the doubly linked list */
    deleteNode(&head, head); /*delete first node*/
    deleteNode(&head, head->next); /*delete middle node*/
    deleteNode(&head, head->next); /*delete last node*/

    /* Modified linked list will be NULL<-8->NULL */
    printf("\n Modified Linked list ");
    printList(head);

    getchar();
}

```

Time Complexity: O(1)

Time Complexity: O(1)

Please write comments if you find any of the above codes/algorithms incorrect, or find better ways to solve the same problem.

705



Subscribe

Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 49 minutes ago

Aman Hi, Why arent we checking for conditions...

Write a C program to Delete a Tree. · 1 hour ago

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 1 hour ago

Sanjay Agarwal bool

tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 1 hour ago

GOPI GOPINATH @admin Highlight this

sentence "We can easily...

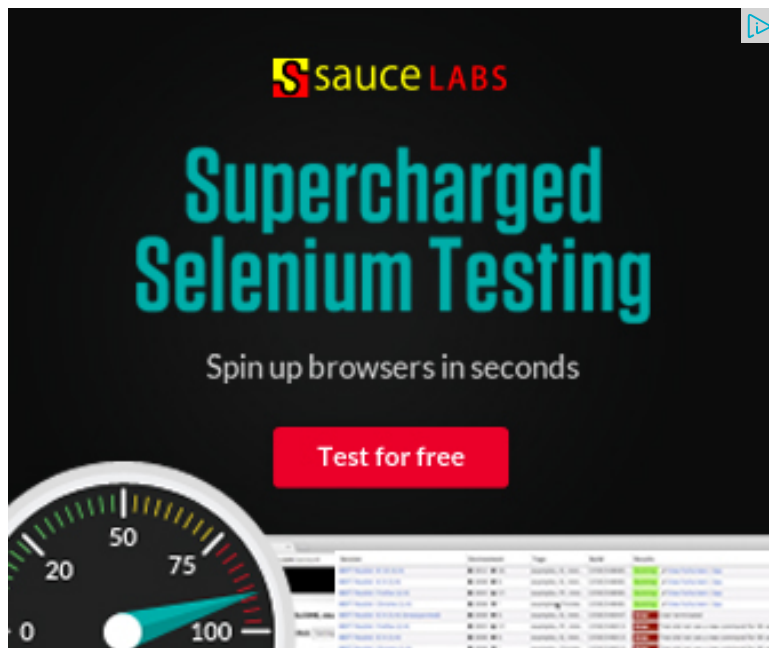
Count trailing zeroes in factorial of a number · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 2 hours ago

AdChoices

[▶ C++ Code](#)



► [Linked Data](#)

AdChoices ►

► [Node 2](#)

► [Programming C++](#)

► [And Node](#)

AdChoices ►

► [Data Node](#)

► [Design Node](#)

► [Java Array](#)

Related Tpoics:

- Given a linked list, reverse alternate nodes and append at the end
- Pairwise swap elements of a given linked list by changing links
- Self Organizing List | Set 1 (Introduction)
- Merge a linked list into another linked list at alternate positions
- QuickSort on Singly Linked List
- Delete N nodes after M nodes of a linked list
- Design a stack with operations on middle element
- Swap Kth node from beginning with Kth node from end in a Linked List



2



Tweet

0



0

Writing code in comment? Please use [ideone.com](#) and share the link here.

10 Comments

GeeksforGeeks

Sort by Newest ▼



Join the discussion



from the discussion...



ronny • 11 months ago

@geeksforgeeks @kartik @venki @sandeep @kartik

In case of deleting the last node, shouldn't the next of second last element be pointing the last element which is being freed. So isn't this a case of DANGLING

Similarly for case of deleting the first node, shouldn't the prev of second element be updated for the same reasons.

This can be done by removing the if conditions in the method.

Correct me if I am wrong.

^ | v • Reply • Share ›



ronny → ronny • 11 months ago

@geeksforgeeks

Sorry for the previous comment.

I misread the function.

So I apologize for any inconvenience hereby caused.

I shall be more careful before posting any comment in future.

3 ^ | v • Reply • Share ›



ashاتم • 11 months ago

when we are making the linked list ourselves and then we need to delete a node that is to be deleted, i.e., how do we get *del?

^ | v • Reply • Share ›



Bala Mastanaiah Yadav • a year ago

theory ple

^ | v • Reply • Share ›



himank • a year ago



minhank · a year ago

i need algo of this

^ | v · Reply · Share ›



sagar patni · 3 years ago

```
#include"dll.h"
```

```
#include
```

```
#include
```

```
void remove1(node *q,dll *l);
```

```
void init(dll *l)
```

```
{
```

```
l->head=l->tail=NULL;
```

```
}
```

```
void append(dll *l ,int x)
```

```
{
```

```
node *p,*q;
```

```
p=(node*)malloc(sizeof(node));
```

```
p->val=x;
```

```
p->next=p->pre=NULL;
```

```
if(l->head==NULL)
```

```
{
```

```
l->head=l->tail=p;
```

[see more](#)

^ | v · Reply · Share ›



sagar patni · 3 years ago

```
/* Paste your code here (You may delete these lines if not writing c)
```

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
void remove1(node *q,dll *l);
```

```
void init(dll *l)
```

```

void init(dll *l)
{
    l->head=l->tail=NULL;
}

void append(dll *l ,int x)
{
    node *p,*q;
    p=(node*)malloc(sizeof(node));
    p->val=x;
    p->next=p->pre=NULL;
    if(l->head==NULL)
    {
        l->head=l->tail=p;
    }
}

```

[see more](#)

^ | v • Reply • Share ›



Mohamed • 3 years ago

Great examples, nicely implemented thank you!

^ | v • Reply • Share ›



Dreamer • 4 years ago

You have not covered the case.. if node is to be deleted is the last node..

Also u could have bit more simplified in terms of cases.. if node to be deleted i

1 Head node

2 Last node

3 Else..

1 ^ | v • Reply • Share ›



Sandeep ➔ Dreamer • 4 years ago

@Dreamer: I believe that case of last node is covered and demonstrat

Could you provide the simplified code that handles all cases?

^ | v • Reply • Share ›

 [Subscribe](#)

 [Add Disqus to your site](#)

@geeksforgeeks, **Some rights reserved**

[Contact Us!](#)

Powered by **WordPress** & **MooTools**, customized by geeksforgeeks team