Deletion in doubly linked list after the specified node

In order to delete the node after the specified data, we need to perform the following steps.

- Copy the head pointer into a temporary pointer temp.

```
temp = head
```

- Traverse the list until we find the desired data value.

```
while(temp -> data != val)
temp = temp -> next;
```

- Check if this is the last node of the list. If it is so then we can't perform deletion.

```
if(temp -> next == NULL)
   {
return;
   }
```

- Check if the node which is to be deleted, is the last node of the list, if it so then we have to make the next pointer of this node point to null so that it can be the new last node of

the list.

```
if(temp -> next -> next == NULL)
   {
      temp ->next = NULL;
   }
```

- Otherwise, make the pointer ptr point to the node which is to be deleted. Make the next of temp point to the next of ptr. Make the previous of next node of ptr point to temp. free the ptr.

```
ptr = temp -> next;
      temp -> next = ptr -> next;
      ptr -> next -> prev = temp;
      free(ptr);
```

Algorithm

- **Step 1:** IF HEAD = NULL

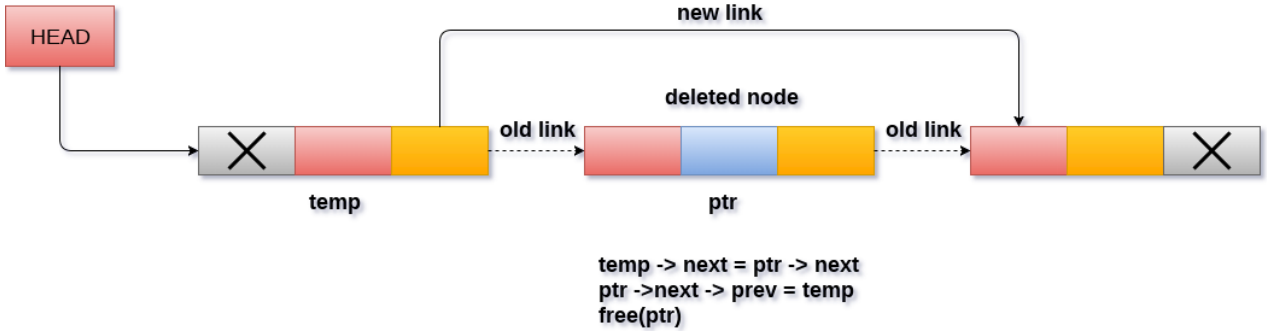
  Write UNDERFLOW
  Go to Step 9
  [END OF IF]

- **Step 2:** SET TEMP = HEAD
- **Step 3:** Repeat Step 4 while TEMP -> DATA != ITEM
- **Step 4:** SET TEMP = TEMP -> NEXT

  [END OF LOOP]

- **Step 5:** SET PTR = TEMP -> NEXT
- **Step 6:** SET TEMP -> NEXT = PTR -> NEXT

- **Step 7:** SET PTR -> NEXT -> PREV = TEMP
- **Step 8:** FREE PTR
- **Step 9:** EXIT



Deletion of a specified node in doubly linked list

C Function

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

```
#include<stdlib.h>
void create(int);
void delete_specified();
struct node
{
   int data;
   struct node *next;
   struct node *prev;
};
struct node *head;
void main ()
{
   int choice,item;
   do
   {
      printf("1.Append List\n2.Delete node\n3.Exit\n4.Enter your choice?");
      scanf("%d",&choice);
      switch(choice)
      {
         case 1:
         printf("\nEnter the item\n");
         scanf("%d",&item);
         create(item);
         break;
         case 2:
         delete_specified();
         break;
         case 3:
         exit(0);
```

```
            break;
            default:
            printf("\nPlease enter valid choice\n");
        }

    }while(choice != 3);
}
void create(int item)
{
   struct node *ptr = (struct node *)malloc(sizeof(struct node));
   if(ptr == NULL)
   {
      printf("\nOVERFLOW\n");
   }
   else
   {


   if(head==NULL)
   {
      ptr->next = NULL;
      ptr->prev=NULL;
      ptr->data=item;
      head=ptr;
   }
   else
   {
      ptr->data=item;
      ptr->prev=NULL;
```

```
      ptr->next = head;
      head->prev=ptr;
      head=ptr;
    }
    printf("\nNode Inserted\n");
}

}
void delete_specified( )
{
    struct node *ptr, *temp;
    int val;
    printf("Enter the value");
    scanf("%d",&val);
    temp = head;
    while(temp -> data != val)
    temp = temp -> next;
    if(temp -> next == NULL)
    {
        printf("\nCan't delete\n");
    }
    else if(temp -> next -> next == NULL)
    {
        temp ->next = NULL;
        printf("\nNode Deleted\n");
    }
    else
    {
        ptr = temp -> next;
```

```
        temp -> next = ptr -> next;
        ptr -> next -> prev = temp;
        free(ptr);
        printf("\nNode Deleted\n");
    }
}
```

Output

```
1.Append List
2.Delete node
3.Exit
4.Enter your choice?1

Enter the item
12

Node Inserted
1.Append List
2.Delete node
3.Exit
4.Enter your choice?1

Enter the item
23

Node Inserted
1.Append List
2.Delete node
3.Exit
4.Enter your choice?1
```

```
Enter the item
34

Node Inserted
1.Append List
2.Delete node
3.Exit
4.Enter your choice?2
Enter the value23

Node Deleted
```

← prev next →

Please Share

Learn Latest Tutorials

Swift

Pig

Flask

C. Graphics

Automata

Testing

Preparation

Aptitude

Reasoning

Verbal A.

Interview

B.Tech / MCA

DBMS

DS

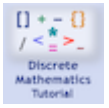
DAA

OS

C. Network

Compiler D.

COA



D. Math.

E. Hacking

Web Tech.

Cyber Sec.

C

C++

Java

.Net

Python

Programs

Control S.