

Deletion in singly linked list after the specified node :

In order to delete the node, which is present after the specified node, we need to skip the desired number of nodes to reach the node after which the node will be deleted. We need to keep track of the two nodes. The one which is to be deleted the other one if the node which is present before that node. For this purpose, two pointers are used: ptr and ptr1.

Use the following statements to do so.

```
ptr=head;
    for(i=0;i<loc;i++)
    {
        ptr1 = ptr;
        ptr = ptr->next;

        if(ptr == NULL)
        {
            printf("\nThere are less than %d elements in the list..",loc);
            return;
        }
    }
```



Now, our task is almost done, we just need to make a few pointer adjustments. Make the next of ptr1 (points to the specified node) point to the next of ptr (the node which is to be deleted).

This will be done by using the following statements.

```
ptr1 ->next = ptr ->next;  
free(ptr);
```

Algorithm

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

```
WRITE UNDERFLOW  
GOTO STEP 10  
END OF IF
```

- **STEP 2:** SET TEMP = HEAD
- **STEP 3:** SET I = 0
- **STEP 4:** REPEAT STEP 5 TO 8 UNTIL I
- **STEP 5:** TEMP1 = TEMP
- **STEP 6:** TEMP = TEMP → NEXT
- **STEP 7:** IF TEMP = NULL

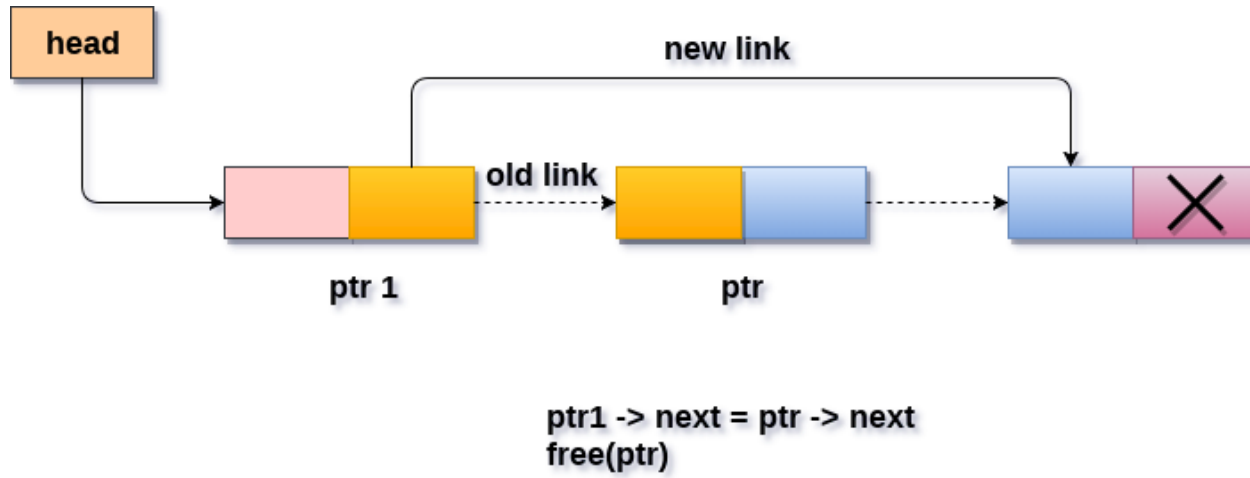
```
WRITE "DESIRED NODE NOT PRESENT"  
GOTO STEP 12  
END OF IF
```

- **STEP 8:** I = I+1

```
END OF LOOP
```

- **STEP 9:** TEMP1 → NEXT = TEMP → NEXT
- **STEP 10:** FREE TEMP
- **STEP 11:** EXIT





Deletion a node from specified position

C function



```
#include<stdio.h>
#include<stdlib.h>
void create(int);
void delete_specified();
struct node
{
    int data;
    struct node *next;
};
struct node *head;
void main ()
{
    int choice,item;
    do
    {
        printf("\n1.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
    {
        ptr->data = item;
        ptr->next = head;
        head = ptr;
        printf("\nNode inserted\n");
    }
}

void delete_specified()
{
    struct node *ptr, *ptr1;
    int loc,i;
    scanf("%d",&loc);
    ptr=head;
```



```
for(i=0;i<loc;i++)
{
    ptr1 = ptr;
    ptr = ptr->next;

    if(ptr == NULL)
    {
        printf("\nThere are less than %d elements in the list..\n",loc);
        return;
    }
}
ptr1 ->next = ptr ->next;
free(ptr);
printf("\nDeleted %d node ",loc);
}
```

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?2

12

There are less than 12 elements in the list..

1.Append List

2.Delete node

3.Exit

4.Enter your choice?2

1

Deleted 1 node

← 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.



Web Tech.



Cyber Sec.



C



C++



Java



.Net



Python



Programs





Control S.

