Delete nodes which have a greater value on right side

Given a singly linked list, remove all the nodes which have a greater value on right side.

Examples:
a) The list 12->15->10->11->5->6->2->3->NULL should be changed to 15->11->6->3->NULL. Note that 12, 10, 5 and 2 have been deleted because there is a greater value on the right side.

When we examine 12, we see that after 12 there is one node with value greater than 12 (i.e. 15), so we delete 12.
When we examine 15, we find no node after 15 that has value greater than 15 so we keep this node.
When we go like this, we get 15->6->3

b) The list 10->20->30->40->50->60->NULL should be changed to 60->NULL. Note that 10, 20, 30, 40 and 50 have been deleted because they all have a greater value on the right side.

c) The list 60->50->40->30->20->10->NULL should not be changed.

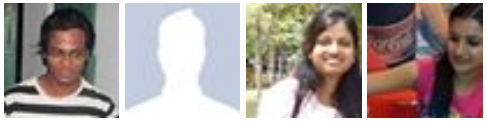
Method 1 (Simple)
Use two loops. In the outer loop, pick nodes of the linked list one by one. In the inner loop, check if there exist a node whose value is greater than the picked node. If there exists a node whose value is greater, then delete the picked node.
Time Complexity: O(n^2)

Method 2 (Use Reverse)
Thanks to Paras for providing the below algorithm.
1. Reverse the list.
2. Traverse the reversed list. Keep max till now. If next node < max, then delete the next node,

Google™ Custom Search

GeeksforGeeks

53,527 people like GeeksforGeeks.



Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

otherwise max = next node.
3. Reverse the list again to retain the original order.

Time Complexity: O(n)

Thanks to R.Srinivasan for providing below code.

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

/* structure of a linked list node */
struct node
{
     int data;
     struct node *next;
};

/* prototype for utility functions */
void reverseList(struct node **headref);
void _delLesserNodes(struct node *head);

/* Deletes nodes which have a node with greater value node
  on left side */
void delLesserNodes(struct node **head_ref)
{
    /* 1) Reverse the linked list */
    reverseList(head_ref);

    /* 2) In the reversed list, delete nodes which have a node
       with greater value node on left side. Note that head
       node is never deleted because it is the leftmost node.*/
    _delLesserNodes(*head_ref);

    /* 3) Reverse the linked list again to retain the
       original order */
    reverseList(head_ref);
}

/* Deletes nodes which have greater value node(s) on left side */
void _delLesserNodes(struct node *head)
{
     struct node *current = head;

     /* Initialize max */
     struct node *maxnode = head;
     struct node *temp;
```

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

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

```
        while (current != NULL && current->next != NULL)
        {
            /* If current is smaller than max, then delete current */
            if(current->next->data < maxnode->data)
            {
                temp = current->next;
                current->next = temp->next;
                free(temp);
            }

            /* If current is greater than max, then update max and
               move current */
            else
            {
                current = current->next;
                maxnode = current;
            }

        }
}

/* Utility function to insert a node at the begining */
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;
}

/* Utility function to reverse a linked list */
void reverseList(struct node **headref)
{
        struct node *current = *headref;
        struct node *prev = NULL;
        struct node *next;
        while(current != NULL)
        {
             next = current->next;
             current->next = prev;
             prev = current;
             current = next;
        }
        *headref = prev;
}
```

Custom market research at scale.

Get $75 off

Google consumer surveys

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

```c
/* Utility function to print a linked list */
void printList(struct node *head)
{
     while(head!=NULL)
     {
        printf("%d ",head->data);
        head=head->next;
     }
     printf("\n");
}

/* Driver program to test above functions */
int main()
{
    struct node *head = NULL;

    /* Create following linked list
      12->15->10->11->5->6->2->3 */
    push(&head,3);
    push(&head,2);
    push(&head,6);
    push(&head,5);
    push(&head,11);
    push(&head,10);
    push(&head,15);
    push(&head,12);

    printf("Given Linked List: ");
    printList(head);

    delLesserNodes(&head);

    printf("\nModified Linked List: ");
    printList(head);

    getchar();
    return 0;
}
```

Output:

```
Given Linked List: 12 15 10 11 5 6 2 3
Modified Linked List: 15 11 6 3
```

Source:

705 Subscribe

Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 43 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 List

▶ Linked Data

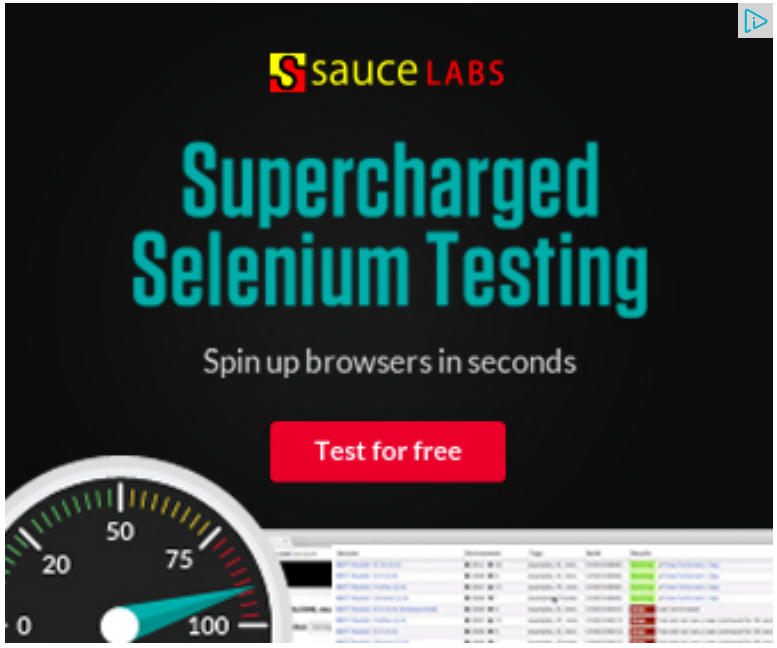
AdChoices

▶ Programming C++

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

http://geeksforgeeks.org/forum/topic/amazon-interview-question-for-software-engineerdeveloper-about-linked-lists-6

Please write comments if you find the above code/algorithm incorrect, or find other ways to solve the same problem.



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

10 Tweet 0 0

Writing code in comment? Please use **ideone.com** and share the link here.

▶ Dom Nodes
AdChoices
▶ Java Array
▶ Java C++
▶ C++ Example

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

@geeksforgeeks, **Some rights reserved** **Contact Us!** Powered by **WordPress** & **MooTools**, customized by geeksforgeeks team

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com