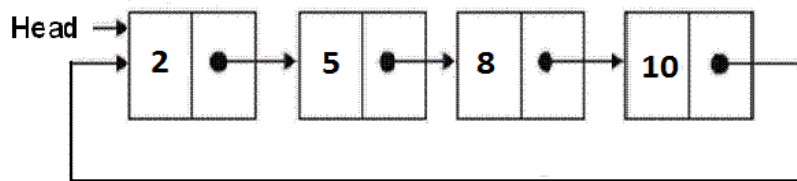


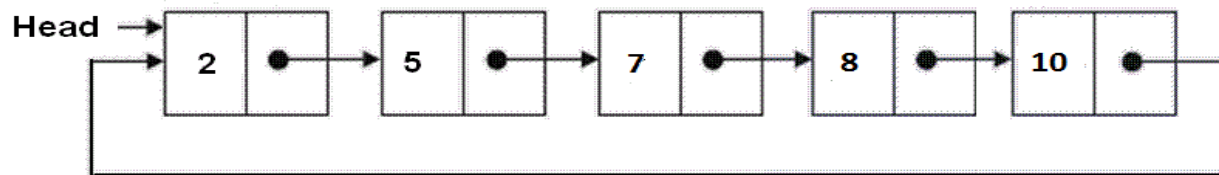
Sorted insert for circular linked list

Difficulty Level: Rookie

Write a C function to insert a new value in a sorted Circular Linked List (CLL). For example, if the input CLL is following.



After insertion of 7, the above CLL should be changed to following



Algorithm:

Allocate memory for the newly inserted node and put data in the newly allocated node. Let the pointer to the new node be new_node. After memory allocation, following are the three cases that need to be handled.

1) *Linked List is empty:*

- since new_node is the only node in CLL, make a self loop.
new_node->next = new_node;
- change the head pointer to point to new node.

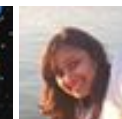
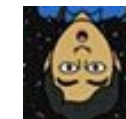
Google™ Custom Search



GeeksforGeeks



53,528 people like GeeksforGeeks.



Interview Experiences

Advanced Data Structures

Dynamic Programming

Greedy Algorithms

Backtracking

Pattern Searching

Divide & Conquer

Mathematical Algorithms

Recursion

Geometric Algorithms

```
*head_ref = new_node;
```

2) *New node is to be inserted just before the head node:*

(a) Find out the last node using a loop.

```
while(current->next != *head_ref)
```

```
    current = current->next;
```

(b) Change the next of last node.

```
    current->next = new_node;
```

(c) Change next of new node to point to head.

```
    new_node->next = *head_ref;
```

(d) change the head pointer to point to new node.

```
    *head_ref = new_node;
```

3) *New node is to be inserted somewhere after the head:*

(a) Locate the node after which new node is to be inserted.

```
while ( current->next!= *head_ref &&
        current->next->data < new_node->data)
```

```
{    current = current->next; }
```

(b) Make next of new_node as next of the located pointer

```
    new_node->next = current->next;
```

(c) Change the next of the located pointer

```
    current->next = new_node;
```

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

```
/* structure for a node */
```

```
struct node
```

```
{
```

```
    int data;
```

```
    struct node *next;
```

```
};
```

```
/* function to insert a new_node in a list in sorted way.
```

```
    Note that this function expects a pointer to head node
```

```
    as this can modify the head of the input linked list */
```

```
void sortedInsert(struct node** head_ref, struct node* new_node)
```

```
{
```

```
    struct node* current = *head_ref;
```

```
    // Case 1 of the above algo
```

```
    if (current == NULL)
```

HP Chromebook 11

 google.com/chromebook

Everything you need in one laptop.
Made with Google. Learn more.



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](#)

```

{
    new_node->next = new_node;
    *head_ref = new_node;
}

// Case 2 of the above algo
else if (current->data >= new_node->data)
{
    /* If value is smaller than head's value then
       we need to change next of last node */
    while(current->next != *head_ref)
        current = current->next;
    current->next = new_node;
    new_node->next = *head_ref;
    *head_ref = new_node;
}

// Case 3 of the above algo
else
{
    /* Locate the node before the point of insertion */
    while (current->next != *head_ref && current->next->data < new_node->data)
        current = current->next;

    new_node->next = current->next;
    current->next = new_node;
}
}

/* Function to print nodes in a given linked list */
void printList(struct node *start)
{
    struct node *temp;

    if(start != NULL)
    {
        temp = start;
        printf("\n");
        do {
            printf("%d ", temp->data);
            temp = temp->next;
        } while(temp != start);
    }
}

/* Driver program to test above functions */
int main()

```

Custom market
research at scale.

Get \$75 off

 Google consumer surveys



```
{
    int arr[] = {12, 56, 2, 11, 1, 90};
    int list_size, i;

    /* start with empty linked list */
    struct node *start = NULL;
    struct node *temp;

    /* Create linked list from the array arr[].
       Created linked list will be 1->2->11->56->12 */
    for(i = 0; i < 6; i++)
    {
        temp = (struct node *) malloc(sizeof(struct node));
        temp->data = arr[i];
        sortedInsert(&start, temp);
    }

    printList(start);
    getchar();
    return 0;
}
```

Output:

1 2 11 12 56 90

Time Complexity: $O(n)$ where n is the number of nodes in the given linked list.

Case 2 of the above algorithm/code can be optimized. Please see [this](#) comment from Pavan. To implement the suggested change we need to modify the case 2 to following.

```
// Case 2 of the above algo
else if (current->data >= new_node->data)
{
    // swap the data part of head node and new node
    swap(&(current->data), &(new_node->data)); // assuming that we have

    new_node->next = (*head_ref)->next;
    (*head_ref)->next = new_node;
}
```

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

Recent Comments

Abhi You live US or India?

[Google \(Mountain View\) interview](#) · 52 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 · 2 hours ago

[GOPI GOPINATH](#) @admin Highlight this sentence "We can easily...

[Count trailing zeroes in factorial of a number](#) · 2 hours ago

[newCoder3006](#) If the array contains negative numbers also. We...

[Find subarray with given sum](#) · 2 hours ago

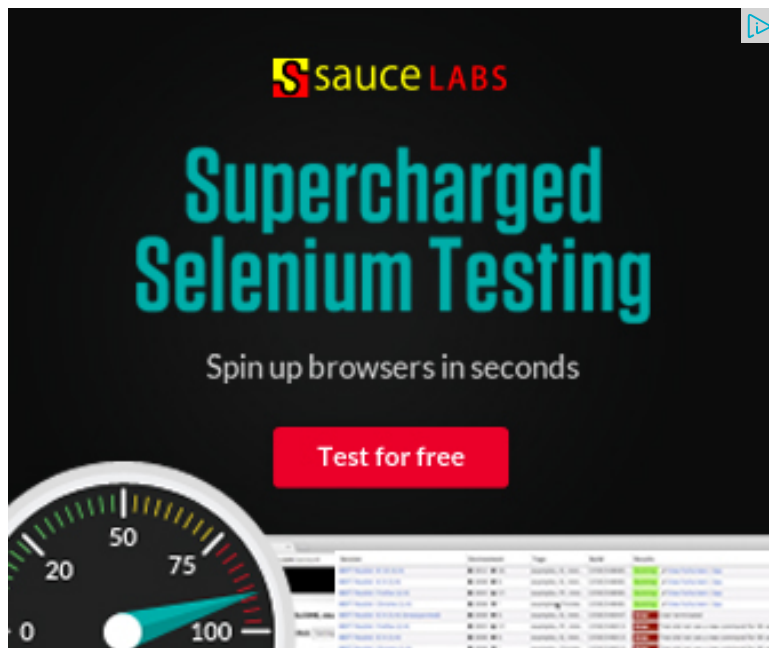
AdChoices 

[► Insertion Insert](#)

[► C++ Insert List](#)

[► Insert Java](#)

AdChoices 



► [Insert Into](#)

► [Sorted](#)

► [Linked List](#)

AdChoices ►

► [Node](#)

► [Insert Function](#)

► [Insert Graph](#)

Related Topics:

- 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.

11 Comments

GeeksforGeeks

Sort by Newest ▼



Join the discussion



with the discussion...



gr81 · 11 months ago

please reply if there is any issue with below code.

test code.

array a is populated with random number.

```
for(int i = 0; i < maxnum; ++i)
```

```
createSortedList(head, a[i]);
```

```
[sourcecode language="C"]
```

```
void createSortedList(NODE *&head, int data)
```

```
{
```

```
if(!head)
```

```
{
```

```
head = getNode(data);
```

```
head->next = head;
```

```
}
```

```
else if( head->data > data)
```

```
{
```

```
NODE *tmp = getNode(data);
```

```
NODE *first = tmp;
```

[see more](#)

^ | v · [Reply](#) · [Share](#) ›



Nishant · 11 months ago

Instead of having a pointer to head node in circular linked it is more convenient because then we can have pointer to the last node as well as the first node. O

```
head = tail->next;
```

Benifits :

- so inserting at the beginning does not require moving in a loop for n times to the length of the linked list.

^ | v • Reply • Share ›



rakitic → Nishant • 10 months ago

nice

```
/* Paste your code here (You may delete these lines if not wri
```

^ | v • Reply • Share ›



Nishant Gaurav • 11 months ago

instead of having a pointer to head node in circular linked it is more convenient because then we can have pointer to the last node as well as the first node. O

head = tail->next;

Benifits :

- so inserting at the beginning does not require moving in a loop for n times to being the length of the linked list.

^ | v • Reply • Share ›



Arindam Sanyal • a year ago

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

```
#include<conio.h>
```

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

```
struct node{
```

```
int info;
```

```
struct node *link;.
```

```
};
```

```
struct node * addtoempty(struct node *, int);.
```

```
struct node * addtoend(struct node *, int);.
```

```
void display(struct node *);.
```

```
void main(){
```

```
clrscr();
struct node *last=NULL;
int num, d;
printf("\n enter the number of nodes...");
scanf("%d",&num);
```

[see more](#)

^ | v • Reply • Share ›



Ashish • 2 years ago

```
int insertCLL(struct node **pCLL, int d)
{
    struct node *pNew= (struct node*)(malloc(sizeof(struct node)));
    if (!pNew) return 0;
    pNew->d=d; pNew->next = NULL;
    struct node *pHead = *pCLL;
    if (!pHead) {
        pNew->next = pNew; *pCLL = pNew; return;
    }
    if (pHead->next == pHead){
        pNew->next = pHead->next;
        pHead->next = pNew;
        *pCLL = pHead; return;
    }

    struct node *pPrev = phead;
```

[see more](#)

^ | v • Reply • Share ›



mms • 2 years ago

thanks

very helpful

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

^ | v • Reply • Share ›



Rajiv Chandel • 3 years ago

Output of the above program should be:-

1 2 11 12 56 90

not

1 2 11 56 12

thank you for the website really helpful.

^ | v • Reply • Share ›



Sandeep → Rajiv Chandel • 3 years ago

@Rajiv Chandel: Thanks for pointing this out. We have corrected it.

^ | v • Reply • Share ›



Pavan • 3 years ago

For the second part (point 2), we need not find the last node to insert the new

1. Copy the data part of first node to new_node->data and write the data to be

2. Insert the new_node in the 2nd position.

^ | v • Reply • Share ›



Sandeep → Pavan • 3 years ago

@pavan: Looks good. We will update the post with suggested optimization

^ | v • Reply • Share ›



Subscribe



Add Disqus to your site

@geeksforgeeks, **Some rights reserved**

Contact Us!

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