

## Pairwise swap elements of a given linked list by changing links

Given a singly linked list, write a function to swap elements pairwise. For example, if the linked list is 1->2->3->4->5->6->7 then the function should change it to 2->1->4->3->6->5->7, and if the linked list is 1->2->3->4->5->6 then the function should change it to 2->1->4->3->6->5

This problem has been discussed [here](#). The solution provided there swaps data of nodes. If data contains many fields, there will be many swap operations. So changing links is a better idea in general. Following is a C implementation that changes links instead of swapping data.

```
/* This program swaps the nodes of linked list rather than swapping the field from the nodes.
Imagine a case where a node contains many fields, there will be plenty of unnecessary swap calls. */
```

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

```
/* A linked list node */
struct node
{
    int data;
    struct node *next;
};
```

```
/* Function to pairwise swap elements of a linked list */
void pairWiseSwap(struct node **head)
{
    // If linked list is empty or there is only one node in list
    if (*head == NULL || (*head)->next == NULL)
        return;
```

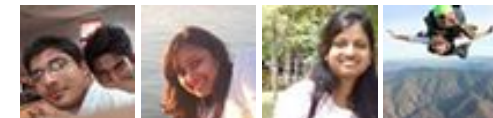
Google™ Custom Search



GeeksforGeeks



53,527 people like [GeeksforGeeks](#).



Facebook

[Interview Experiences](#)

[Advanced Data Structures](#)

[Dynamic Programming](#)

[Greedy Algorithms](#)

[Backtracking](#)

[Pattern Searching](#)

[Divide & Conquer](#)

[Mathematical Algorithms](#)

[Recursion](#)

[Geometric Algorithms](#)

```

// Initialize previous and current pointers
struct node *prev = *head;
struct node *curr = (*head)->next;

*head = curr; // Change head before proceeding

// Traverse the list
while (true)
{
    struct node *next = curr->next;
    curr->next = prev; // Change next of current as previous node

    // If next NULL or next is the last node
    if (next == NULL || next->next == NULL)
    {
        prev->next = next;
        break;
    }

    // Change next of previous to next next
    prev->next = next->next;

    // Update previous and curr
    prev = next;
    curr = prev->next;
}

}

/* Function to add a node at the beginning of 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;

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

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

/* Function to print nodes in a given linked list */

```



## 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

```

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

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

    /* The constructed linked list is:
    1->2->3->4->5->6->7 */
    push(&start, 7);
    push(&start, 6);
    push(&start, 5);
    push(&start, 4);
    push(&start, 3);
    push(&start, 2);
    push(&start, 1);

    printf("\n Linked list before calling pairWiseSwap() ");
    printList(start);

    pairWiseSwap(&start);

    printf("\n Linked list after calling pairWiseSwap() ");
    printList(start);

    getchar();
    return 0;
}

```

Output:

```

Linked list before calling pairWiseSwap() 1 2 3 4 5 6 7
Linked list after calling pairWiseSwap() 2 1 4 3 6 5 7

```

Time Complexity: Time complexity of the above program is  $O(n)$  where  $n$  is the number of nodes in a given linked list. The while loop does a traversal of the given linked list.

Following is **recursive implementation** of the same approach. We change first two nodes and



recur for the remaining list. Thanks to geek and omer salem for suggesting this method.

```
/* This program swaps the nodes of linked list rather than swapping the  
field from the nodes.  
Imagine a case where a node contains many fields, there will be plenty  
of unnecessary swap calls. */
```

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

```
/* A linked list node */  
struct node  
{  
    int data;  
    struct node *next;  
};
```

```
/* Function to pairwise swap elements of a linked list.  
It returns head of the modified list, so return value  
of this node must be assigned */
```

```
struct node *pairWiseSwap(struct node* head)  
{  
    // Base Case: The list is empty or has only one node  
    if (head == NULL || head->next == NULL)  
        return head;  
  
    // Store head of list after two nodes  
    struct node* remaing = head->next->next;  
  
    // Change head  
    struct node* newhead = head->next;  
  
    // Change next of second node  
    head->next->next = head;  
  
    // Recur for remaining list and change next of head  
    head->next = pairWiseSwap(remaing);  
  
    // Return new head of modified list  
    return newhead;  
}
```

```
/* Function to add a node at the beginning of Linked List */  
void push(struct node** head_ref, int new_data)  
{  
    /* allocate node */
```

## Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 47 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 

[▶ Linked List](#)

[▶ C++ Code](#)


[▶ Programming C++](#)

AdChoices 

[▶ C++ Swap String](#)

► [Java Array](#)

► [Null Pointer](#)

AdChoices 

► [C++ Example](#)

► [C++ Program](#)

► [Test C++](#)

```
struct node* new_node =
    (struct node*) malloc(sizeof(struct node));

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

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

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

/* Function to print nodes in a given linked list */
void printList(struct node *node)
{
    while(node != NULL)
    {
        printf("%d ", node->data);
        node = node->next;
    }
}

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

    /* The constructed linked list is:
    1->2->3->4->5->6->7 */
    push(&start, 7);
    push(&start, 6);
    push(&start, 5);
    push(&start, 4);
    push(&start, 3);
    push(&start, 2);
    push(&start, 1);

    printf("\n Linked list before calling pairWiseSwap() ");
    printList(start);

    start = pairWiseSwap(start); // NOTE THIS CHANGE

    printf("\n Linked list after calling pairWiseSwap() ");
    printList(start);

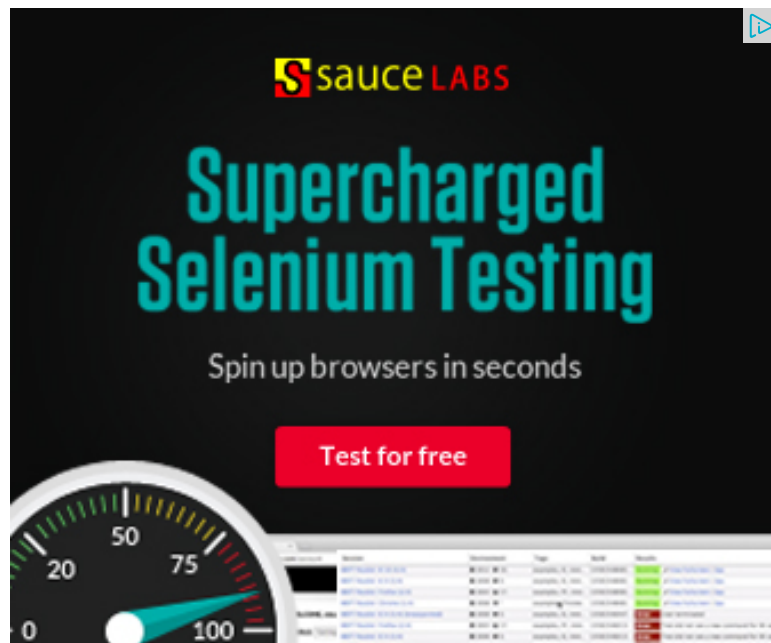
    return 0;
}
```

```
}
```

Linked list before calling pairWiseSwap() 1 2 3 4 5 6 7

Linked list after calling pairWiseSwap() 2 1 4 3 6 5 7

This article is contributed by **Gautam Kumar**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above



## Related Tpoics:

- Given a linked list, reverse alternate nodes and append at the end
- 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
- QuickSort on Doubly Linked List



12



Tweet 3



2

**Writing code in comment?** Please use [ideone.com](https://ideone.com) and share the link here.

@geeksforgeeks, **Some rights reserved**

**Contact Us!**

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