Write a recursive function to print reverse of a Linked List

Note that the question is only about printing the reverse. To reverse the list itself see this

Difficulty Level: Rookie

Algorithm

```
printReverse(head)
  1. call print reverse for hed->next
  2. print head->data
```

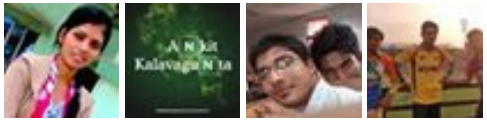
Implementation:

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

/* Link list node */
struct node
{
    int data;
    struct node* next;
};

/* Function to reverse the linked list */
void printReverse(struct node* head)
{
  // Base case
  if(head == NULL)
    return;

  // print the list after head node
```

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

```
  printReverse(head->next);

  // After everything else is printed, print head
  printf("%d  ", head->data);
}

/*UTILITY FUNCTIONS*/
/* Push a node to linked list. Note that this function
  changes the head */
void push(struct node** head_ref, char 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 pochar to the new node */
    (*head_ref)    = new_node;
}

/* Drier program to test above function*/
int main()
{

  struct node* head = NULL;

  push(&head, 1);
  push(&head, 2);
  push(&head, 3);
  push(&head, 4);

  printReverse(head);
  getchar();
}
```

Time Complexity: O(n)

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

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

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

8 Comments GeeksforGeeks

Sort by Newest

Join the discussion…

Codeguru • 5 months ago

complete c code to reverse a linked list

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

struct node
{
int data;
struct node * link;
};

void insert_beg(struct node ** head,int num)
{
struct node * temp;
temp=(struct node *)malloc(sizeof(struct node));
temp->data=num;
temp->link=*head;
*head=temp;
return;
```

see more

2 ▲ | ▼ • Reply • Share ›

prathviraj • 10 months ago

Could you give a solution without using recursive function?

▲ | ▼ • Reply • Share ›

Abhay ➚ prathviraj • 6 months ago

/* ya linklist can be reversed using stack or simply array*/

705 Subscribe

Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 49 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

list. After theat push element on to the stack, after pushing all the elem
doing this you can pring the linklist in reverse order.
the simple array code is given below

```
i=0;

while(ptr->link!=NULL)
{
arr[i]=ptr->info;
ptr=ptr->link;
i++;

}
for(j=i-1;j>=0;j--)
printf("%d",arr[j]);
```

2 ▲ | ▼ · Reply · Share ›

Anon ➦ prathviraj · 7 months ago

will be sort of cheating but you can use a stack ;)

▲ | ▼ · Reply · Share ›

Scholastica Peter · a year ago

so good when using recursive.

▲ | ▼ · Reply · Share ›

Ashish Singh · a year ago

solve above prblm by using recursive and non recursive function.

▲ | ▼ · Reply · Share ›

code4fun · 2 years ago

Here it is in C

```
typedef struct node
{
        int data;
        struct node* next;
}NODE,*PNODE;

PNODE reverseLinkList(PNODE head)
{
        PNODE temp ;
        if (!head || !head->next)
                return head;

        temp = reverseLinkList(head->next);
        head->next->next = head;
        head->next = NULL;
        return temp;
}
```


BlackMath • 2 years ago

A java program to actually reverse a linked list by recursion, not only printing it.

```
/* Paste your code here (You may delete these lines if not writing co
class LNode
{
   int value;
   LNode next;

   LNode (int val)
   {
```

```
        value = val;
    }
}

public class ReverseLinkedListByRecursion
{
  public static void printList (LNode head)
```

see more

1 ▲ | ▼ • Reply • Share ›

Subscribe Add Disqus to your site