Reverse words in a given string

Example: Let the input string be “i like this program very much”. The function should change the string to “much very program this like i”

Algorithm:

```
1) Reverse the individual words, we get the below string.
     "i ekil siht margorp yrev hcum"
2) Reverse the whole string from start to end and you get the desired output.
     "much very program this like i"
```

```
#include<stdio.h>

/* function prototype for utility function to
  reverse a string from begin to end  */
void reverse(char *begin, char *end);

/*Function to reverse words*/
void reverseWords(char *s)
{
  char *word_begin = s;
  char *temp = s; /* temp is for word boundry */

  /*STEP 1 of the above algorithm */
  while( *temp )
  {
    temp++;
    if (*temp == '\0')
    {
      reverse(word_begin, temp-1);
    }
    else if(*temp == ' ')
    {
```

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

```
      reverse(word_begin, temp-1);
      word_begin = temp+1;
    }
  } /* End of while */

   /*STEP 2 of the above algorithm */
  reverse(s, temp-1);
}

/* UTILITY FUNCTIONS */
/*Function to reverse any sequence starting with pointer
  begin and ending with pointer end  */
void reverse(char *begin, char *end)
{
  char temp;
  while (begin < end)
  {
    temp = *begin;
    *begin++ = *end;
    *end-- = temp;
  }
}

/* Driver function to test above functions */
int main()
{
  char s[] = "i like this program very much";
  char *temp = s;
  reverseWords(s);
  printf("%s", s);
  getchar();
  return 0;
}
```

The above code doesn't handle the cases when the string starts with space. The following version handles this specific case and doesn't make unnecessary calls to reverse function in the case of multiple space in between. Thanks to rka143 for providing this version.

```
void reverseWords(char *s)
{
    char *word_begin = NULL;
    char *temp = s; /* temp is for word boundry */

    /*STEP 1 of the above algorithm */
    while( *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

```
        /*This condition is to make sure that the string start with
          valid character (not space) only*/
        if (( word_begin == NULL ) && (*temp != ' ') )
        {
            word_begin=temp;
        }
        if(word_begin && ((*(temp+1) == ' ') || (*(temp+1) == '\0')))
        {
            reverse(word_begin, temp);
            word_begin = NULL;
        }
        temp++;
    } /* End of while */

    /*STEP 2 of the above algorithm */
    reverse(s, temp-1);
}
```

Time Complexity: O(n)

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

Related Tpoics:

- Printing Longest Common Subsequence
- Suffix Array | Set 2 (nLogn Algorithm)
- Rearrange a string so that all same characters become d distance away
- Recursively remove all adjacent duplicates
- Find the first non-repeating character from a stream of characters
- Dynamic Programming | Set 33 (Find if a string is interleaved of two other strings)
- Remove “b” and “ac” from a given string
- Dynamic Programming | Set 29 (Longest Common Substring)

15 Tweet 0 0

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

71 Comments GeeksforGeeks

Sort by Newest ▾

Join the discussion…

Abhishek Kumar • 18 days ago

can we do like this..??
1)first store d whole string in an stack..
2)after that pop the words from stack nd wenever we meet with space we just

4 ^ | ˅ • Reply • Share ›

VaraKalyan M • 19 days ago

Can be solved easily using stack in 0(n)

2 ^ | ˅ • Reply • Share ›

695 Subscribe

Recent Comments

affiszerv Your example has two 4s on row 3, that's why it...

Backtracking | Set 7 (Sudoku) · 20 minutes ago

RVM Can someone please elaborate this Qs from above...

Flipkart Interview | Set 6 · 40 minutes ago

Vishal Gupta I talked about as an Interviewer in general,...

Software Engineering Lab, Samsung Interview | Set 2 · 40 minutes ago

@meya Working solution for question 2 of 4f2f round....

Amazon Interview | Set 53 (For SDE-1) · 1 hour ago

sandeep void rearrange(struct node *head) {...

Given a linked list, reverse alternate nodes and append at the end · 2 hours ago

Neha I think that is what it should return as, in...

Find depth of the deepest odd level leaf node · 2 hours ago

poorvisha · 24 days ago

.if i/p is :my< double space >name< single space >khan

o/p will be:khan< single space >name< double space >my , according to the a

but o/p should be :khan< double space >name< single space >my.

am i correct? the above case is not handled.

1 ^ | ˅ · Reply · Share ›

Danie · a month ago

string reversing in java :

kodingexamples.blogspot.com/20...

^ | ˅ · Reply · Share ›

Mukesh Sharma · a month ago

```
public static String reverseWords(String s)
{
String str="";
String fin="";

for(int i=s.length()-1;i>=0;i--)

{

str=str+s.charAt(i);

}

int beg=0;

for(int i=0;i<str.length();i++) {="" if((str.charat(i)="=" '))="" {="" for(int="" j="i-1;j
```

{

fin=fin+str.charAt(j);

see more

^ | ⌄ • Reply • Share ›

pavansrinivas • 7 months ago

How can I reduce my space complexity??

```
void reverseWords(){
     String a = "a efgh klmnop";
     String res = "";
     String res2 = "";
     int  i =0;
     while(i<a.length()){ int="" j="i;" while(i<a.length()&&a.chara
             res = res+a.charAt(k);
             k--;
         }
         res = res+" ";
         i++;
     }
     for(int l=a.length()-1;l>=0;l--){
         res2+=res.charAt(l);
     }
     System.out.print(res2);
 }
```

^ | ⌄ • Reply • Share ›

lamateur • 7 months ago

Hey somebody please give me the right program for this..So many replies whi

^ | ⌄ • Reply • Share ›

Charm03 ➦ lamateur • 3 months ago

I agree, someone please rate which one is most optimal and correct a

^ | ⌄ • Reply • Share ›

surbhi • 9 months ago

```
#include<stdio.h>
int main()
{

char s[100],s1[100];
int i,j=0,l=0;
printf("enter any string\n");
fgets(s,100,stdin);

for(i=0;s[i]!='\0';i++)
l++;

for(i=l;i>=0;i--)
s1[j++]=s[i];

s1[j]='\0';
printf("string is \n":);
fputs(s1,stdout);

return 0;
}
```

2 ^ | ⌄ • Reply • Share ›

Karan • 9 months ago

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

```
void push( char stack[], char c,int* top,int len )
{
        if((*top) == len)
            return;
        stack[(*top)++]=c;
}

char pop( char stack[],int* top )
{
    (*top)--;
    return stack[*top];
}
```

see more

^ | ⌄ • Reply • Share ›

Karan • 9 months ago

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


void push( char stack[], char c,int* top,int len )
{
        if((*top) == len)
            return;
        stack[(*top)++]=c;
}
```

```
char pop( char stack[],int* top )
{
    (*top)--;
    return stack[*top];
}
```

see more

^ | ⌄ • Reply • Share ›

Vignesh • 9 months ago

```
# include
# include
void strnrev(char *s,int n)
{
if(n < 0)
return;
printf("%c",s[n]);
strnrev(s,n-1);
}

int main()
{
char s[50];
gets(s);
strnrev(s,strlen(s)-1);
getch();
}
```

^ | ⌄ • Reply • Share ›

Akash Tiwari • 9 months ago

#include<stdio.h>

```
#include<stdio.h>
#include<conio.h>
#define max 10.
void revert(int i, int j, char str[]);.
void main()
{
char str[]="this is a good boy";.
int i, len, start=0;.
len=strlen(str);.
strrev(str);.
// printf("%s", str);.
for(i=0;i<len;i++).
{.

if(str[i]==&#039 &#039)
{
revert(start, i-1, str);
start=i+1;
```

see more

⌃ | ⌄ • Reply • Share ›

ritesh • 9 months ago

Reversing words in a string.

```
#include
#include "conio.h"

using namespace std;

void reverse_string( char * beg,char * end)
{
while((beg<end))
{ char temp=*beg;
*beg=*end;
*end=temp;
beg++;end--;
}
}
void reverse_string_word(char *str)
{ char *beg1=str;
char *end1=str;
while(*end1)
```

see more

^ | ⌄ • Reply • Share ›

aspire • 9 months ago

@GeeksforGeeks : The complexity of the above code looks like O(n2) becaus
traversing it again for reversing purpose. Please look into this.

1 ^ | ⌄ • Reply • Share ›

Sashank Swaminatan • 10 months ago

dude , u r using huge space
if space is given , we can read array in rev orderand use a stack to push in ev
encountered , and print

^ | ⌄ • Reply • Share ›

pratul • 10 months ago

The code I've written below is pretty basic and not much complicated !!! ...

```
#include
#include
void fun(char a[])
{
int k,j,l=strlen(a),p=l;
for(k=l-1;k>=0;k--)
{
if(a[k]==32 || k==0)
{
j=(k==0)?0:k+1;
for(;j<p;j++)
printf("%c",a[j]);

printf(" ");
p=k;
}
}
}
int main()
{
char str[] = "i am a good boy";
fun(str);
return 0;
}
```

Reply • Share ›

raghavendar pratul • 9 months ago

pratul please give me explanation for your prigram

Reply • Share ›

pratul raghavendar • 9 months ago

Hi... what I'm doing in my code is that I'm scanning the whole s
encounter a space (or the string reaches its starting)..I reverse

For this purpose,I have maintained a variable 'p' which is updat
purpose.

Just go through the code once...u'll get to know how it actually \

```
/* Paste your code here (You may delete these lines if
```

^ | ⌄ • Reply • Share ›

Aditya Ambashtha • 10 months ago

This is compact and works well i guess...

```
 void printReverse(char *str)
{
    char temp[10];
    int cnt;

    for(cnt=0;(str[cnt]!=' ')&&(str[cnt]!='&#092&#048');cnt++)
        temp[cnt]=str[cnt];



    temp[cnt]='&#092&#048';



    if (str[cnt]!='&#092&#048')
        printReverse(&str[cnt+1]);

    printf("%s ",temp);
}
```

^ | ⌄ • Reply • Share ›

Ashish Kr Singh • 10 months ago

#include <stdio.h>
#include <iostream>
#include<string.h>
using namespace std;

```
using namespace std;.
void reverse(char *str).
{
int len = strlen(str);.
int I =0;.
char *start=str, * end= str+len-1;.
while(end>=start).
{.

char temp = *start;.

*start = *end;.

*end=temp;.

start++;.
```

see more

^ | ⌄ • Reply • Share ›

Prateek Sharma • a year ago

Python code with o(n) time and o(1) space complexity...

```
[sourcecode language="Python"]
def swap(list1,j,i):
if j >= i:
return 0
else:
temp = list1[j]
list1[j] = list1[i]
list1[i] = temp
swap(list1,j+1,i-1)
def reverseStringWordsPosition(s):
list1 = list(s)
```

```
j =0
for i in range(len(list1)):
if list1[i] !=" ":
continue
elif i ==0 or list1[i-1] == " " :
```

see more

1 ⌃ | ⌄ • Reply • Share ›

abhishek08aug • a year ago

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

void reverse_string(char * start, char * end) {
  char * temp=(char *)malloc(sizeof(char));
  while(start<end) {
    *temp=*start;
    *start=*end;
    *end=*temp;
    start++;
    end--;
  }
}

void reverse_words(char * str) {
  char * word_start=str;
  if(word_start==NULL) {
```

see more

⌃ | ⌄ • Reply • Share ›

abhishek08aug ➜ abhishek08aug · a year ago

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

void reverse_string(char * start, char * end) {
char * temp=(char *)malloc(sizeof(char));
while(start<end) {="" *temp="*start;" *start="*end;" *end="*temp;" start
reverse_words(char="" *="" str)="" {="" char="" *="" word_start="str;" i
}="" char="" *="" temp="str;" while(*temp!="\0" )="" {="" temp++;="" if(*
reverse_string(word_start,="" temp-1);="" word_start="temp+1;" }="" e
reverse_string(word_start,="" temp-1);="" }="" }="" reverse_string(str,=
char="" *="" str="(char" *)malloc(sizeof(char)*40);="" strcpy(str,="" "yo
much");="" reverse_words(str);="" printf("string="" after="" reversing="
return="" 0;="" }="" string="" after="" reversing="" words="" is:="" much
you="">
```

Reply · Share ›

pranav · a year ago

```
void reverse(char s[],int low,int high)
{
        int i,j;
        int temp;

        for (i=low,j=high;i<j;i++,j--)
        {
                temp = s[i];
                s[i] = s[j];
                s[j] = temp;
        }
}
```

```
#define DELIM 1
#define NO_DELIM 0

void reverseWords(char s[])
```

see more

⌃ | ⌄ • Reply • Share ›

Shankar Cool • a year ago

shortest of all....

```
void main(){
int i=0, j=0;
char **r, s;
clrscr();
while((s=getchar())!=&#039n&#039)
{
if(s==&#039 &#039) { r[i][j++]=&#039&#039; j=0; ++i;}.
else r[i][j++]=s;
}
r[i][j++]=&#039&#039;
while(i>=0)
printf("%s ", r[i--]);
}
```

⌃ | ⌄ • Reply • Share ›

Star_Trek • a year ago

```
reverse_words_in_a_string(string,stlength)
{
for(start=end=0;end<stlength;end++)
{
if(string[end]!=&#039&#039)
{
start=end;
while(string[end]!=&#039&#039 && end<stlength)
{
end++;
}
end--;
reverseword(string,start,end);
}
}
}
```

reverse string(string,strstart,strend)

see more

^ | ⌄ • Reply • Share ›

RGUIIIT • a year ago

```
 #include<stdio.h>
#include<string.h>
main()
{
      char *pch;
      char str[]= "This is a sample string.";
      printf("Reversing the sentence \"%s\"\n",str);
      pch = strtok(str,"\n");

      while(pch != NULL)
      {
                strrev(pch);
                printf("\n%s\n",pch);
                pch = strtok(NULL, "\n");
```

```
        }
        getch();
}
```

⌃ | ⌄ • Reply • Share ›

Gangadhara Pagadala • a year ago

```
#include<stdio.h>
#include<string.h>
char s[]="I love India";.
void print(int beg, int end).
{
while(beg!=end)
printf("%c", s[++beg]);
}
int main()
{
int i=strlen(s), j=i-1;
while(i--)
{
if(s[i]==&#039 &#039).
{.
print(i, j);.
j=i-1;.
}.
}.
return 0;.
}
```

⌃ | ⌄ • Reply • Share ›

anand • a year ago

[sourcecode language="java"]

```
[sourcecode language="java"]
/* private static String reverseAText(String s) {
String[] sarray = s.split(" ");
return reverseAText(sarray, 0 , sarray.length - 1);
}

private static String reverseAText(String[] s,int i ,int n) {
if(i == n)
return s[n];
else{
return reverseAText(s, i+1, n)+" "+s[i];
}
} */
```

^ | ⌄ • Reply • Share ›

Gangadhara • a year ago

```
#include<stdio.h>
#include<string.h>
char str[]="I love india";
void Reverse(char *revStr)
{
        if(!revStr)return;
        Reverse(strtok(NULL," "));
        printf("%s ",revStr);
}
int main()
{
        Reverse(strtok(str," "));
        return 0;
}
```

^ | ⌄ • Reply • Share ›

^ | v · Reply · Share ›

Gangadhara · a year ago

```
#include
#include
char str[]="I love india";
void Reverse(char *revStr)
{
if(!revStr)return;
Reverse(strtok(NULL," "));
printf("%s ",revStr);
}
int main()
{
Reverse(strtok(str," "));
return 0;
}
```

1 ^ | v · Reply · Share ›

Marsha Donna ➦ Gangadhara · 3 months ago

can u pls explain wat does Reverse(strtok(NULL," ")); do??

^ | v · Reply · Share ›

Vibhu Tiwari ➦ Gangadhara · a year ago

great solution man..

^ | v · Reply · Share ›

sush · a year ago

Simple and easy implementation

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

```
#include<string.h>
/*Function to reverse words*/
void reverse(char* str,int l,int h)
{
    if(l<=h)
        {
                char temp=str[l];
                str[l]=str[h];
                str[h]=temp;
                reverse(str,l+1,h-1);
        }
}
void reverse_words(char* s)
{
```

see more

^ | ⌄ • Reply • Share ›

datta • 2 years ago

/*You can also tokenize the given string using strtok() and get the string revers

```
#include
#include
#include
#include

void rev(char *start,char *end);
void reverse_whole(char *str);
int main()
{
char *str;
str = malloc(sizeof(char *));
```

```
printf("Enter the string:\n");//Suppose the input is"my name is datta"
// gets(str);
scanf("%[^\n]",str);
reverse_whole(str);
```

see more

1 ^ | ⌄ • Reply • Share ›

pr6989 • 2 years ago

```
 //Reverse the words of a string in place
#include<iostream>
#include<string.h>
#define max 100
using namespace std;
void swap(char*,char*);
int main()
{
    char a[max];
    cout<<"Enter a string : ";
    cin.getline(a,max);
    int i,j,k;
    int count_blanks=0;
    for(i=0;a[i]!='&#092&#048';i++)
    {
        if(a[i]==' ')
        count_blanks++;
    }
```

see more

^ | ⌄ • Reply • Share ›

bhavesh • 2 years ago

```
 #include<stdio.h>
#include<stdlib.h>
#include<string.h>
int main()
{
        char str[]="my  name is bhavesh";
        struct node{
        char  *a;
        struct node *next;
        };
        char *ab;
        struct node *head=(struct node*)malloc (sizeof(struct node));
        ab=strtok(str , " ");
                char *abc=(char *)malloc (sizeof (char)*90);
                strcpy(abc,ab);
                head->a = abc;
                head->next=NULL;
        while((ab=strtok(NULL , " "))!=NULL)
```

see more

^ | ˅ • Reply • Share ›

Geek1 • 2 years ago

Made some changes, like passing temp+1 to reverse, this is a working code.

```
 #include "StdAfx.h"
#include "conio.h"
#include "iostream"
#include "string"
#include "math.h"
using namespace std;
void reverseWords(char *s)
```

```
{
        char *word_begin = NULL;
        char *temp = s; /* temp is for word boundry */

        /*STEP 1 of the above algorithm */
        while( *temp )
        {
                /*This condition is to make sure that the string star
                        valid character (not space) only*/
```

see more

^ | ⌄ • Reply • Share ›

Geek1 • 2 years ago

This one is a working code which would cater the spaces as well, done some some temp+1 is now being passed to the reverse.

```
 #include "StdAfx.h"
#include "conio.h"
#include "iostream"
#include "string"
#include "math.h"
using namespace std;
void reverseWords(char *s)
{
    char *word_begin = NULL;
    char *temp = s; /* temp is for word boundry */

    /*STEP 1 of the above algorithm */
    while( *temp )
    {
        /*This condition is to make sure that the string start with
```

see more

^ | ⌄ • Reply • Share ›

Avinash • 2 years ago

```
 Reverse Words in a sentence

void ReverseWords(char *str)
{
        int wordstart=0, wordend=0;
        int len= strlen(str);
        ReverseString(str,0, len-1)

        for(wordend=0;wordend<len;wordend++)
        {
                If(str[wordend]!='')
                {
                        wordstart=wordend;
                        while(str[wordend]!=''&&wordend<len)
                        {
                                wordend++;
                        }
                        wordend--;
```

see more

^ | ⌄ • Reply • Share ›

Punit • 2 years ago

Following is an implementation in Java using StringBuilder..

[sourcecode language="JAVA"]
public class REVERSEWORDS {
public static void main(String[] args) {

```
public static void main(String[] args) {
StringBuilder sb = new StringBuilder("GEEKS FOR HELP");
System.out.println(sb.toString());
reverseWords(sb);
System.out.println(sb.toString());
}

private static void reverseWords(StringBuilder sb) {
//Step 1 : Reverse individual words
short start_word = 0;
for(int i=0;i<sb.length();i++){
if(sb.charAt(i)==' '){
reverseWord(sb, start_word, i-1);
start_word = (short) (i+1);
}else if(i == sb.length()-1){
```

see more

^ | v • Reply • Share ›

Saurya • 3 years ago

Please comment on my implementation.

```
 /* Paste your code here (You may delete these lines if not writing c
/*Reverse the words in a string*/
#include <iostream>
#include <stack>
using namespace std;
int main()
{
  stack<char> wrd;
  char ar[] =  {"i ekil siht margorp yrev hcum"};
  int i = 0, j=0;
  while(ar[i]!='&#092&#048')
  {
    wrd.push(ar[i]);
    i++;
  }
  i=0;
```

see more

^ | ˅ • Reply • Share ›

topcoder ➦ Saurya • a year ago

Using in-built stack is not allowed in PI's

^ | ˅ • Reply • Share ›

Anonymous • 3 years ago

In case of java, we can do this:
1) Use StringTokenizer to separate the words
2) Push each token into to a stack
3) Pop them out

^ | ˅ • Reply • Share ›

CaesiumX • 3 years ago

Hello geeks team
what about this ->

#include<stdio.h>
#include<conio.h>
#include<string.h>
void reverse(char str[])
{
char *beg=str;

```
char *end=str+strlen(str)-1;
char temp=NULL;
while(beg<end)
{
temp=*beg;
*beg=*end;
*end=temp;
```

see more

^ | ⌄ • Reply • Share ›

CaesiumX ➦ CaesiumX • 3 years ago

@geeksforgeek guys a nice portal like this shouldn't dawdle , on both f
web pages takes a hell lot of time , i know a lot process operations/inte
while n number of users post/code simultaneously but i have seen this
not good considering the type of members we have here.

^ | ⌄ • Reply • Share ›

Sandeep ➦ CaesiumX • 3 years ago

@CaesiumX: The site is currently hosted on a shared web serv
increase in traffic. We will soon be migrating the site to a dedica

^ | ⌄ • Reply • Share ›

amit khoth • 3 years ago

@geeksforgeeks correct this code.for same type of word it is not working.

^ | ⌄ • Reply • Share ›

GeeksforGeeks ➦ amit khoth • 3 years ago

@amit khoth: Could you please provide a sample string for which it did

^ | ⌄ • Reply • Share ›

Load more comments