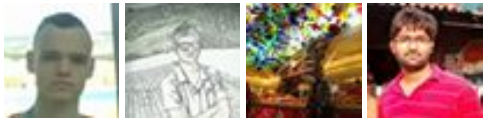
Write your own atoi()

The atoi() function takes a string (which represents an integer) as an argument and returns its value.

Following is a simple implementation. We initialize result as 0. We start from the first character and update result for every character.

```
// A simple C++ program for implementation of atoi
#include <stdio.h>

// A simple atoi() function
int myAtoi(char *str)
{
    int res = 0; // Initialize result

    // Iterate through all characters of input string and update resul
    for (int i = 0; str[i] != '\0'; ++i)
        res = res*10 + str[i] - '0';

    // return result.
    return res;
}

// Driver program to test above function
int main()
{
    char str[] = "89789";
    int val = myAtoi(str);
    printf ("%d ", val);
    return 0;
}
```

Output:



```
89789
```

The above function doesn't handle negative numbers. **Following is a simple extension to handle negative numbers**.

```
// A C++ program for implementation of atoi
#include <stdio.h>

// A simple atoi() function
int myAtoi(char *str)
{
    int res = 0;  // Initialize result
    int sign = 1;  // Initialize sign as positive
    int i = 0;  // Initialize index of first digit

    // If number is negative, then update sign
    if (str[0] == '-')
    {
        sign = -1;
        i++;  // Also update index of first digit
    }

    // Iterate through all digits and update the result
    for (; str[i] != '\0'; ++i)
        res = res*10 + str[i] - '0';

    // Return result with sign
    return sign*res;
}

// Driver program to test above function
int main()
{
    char str[] = "-123";
    int val = myAtoi(str);
    printf ("%d ", val);
    return 0;
}
```

Output:

```
-123
```

The above implementation doesn't handle errors. What if *str* is NULL or *str* contains non-numeric characters. **Following implementation handles errors**.

Popular Posts

```
// A simple C++ program for implementation of atoi

#include <stdio.h>

// A utility function to check whether x is numeric
bool isNumericChar(char x)
{
    return (x >= '0' && x <= '9')? true: false;
}

// A simple atoi() function. If the given string contains
// any invalid character, then this function returns 0
int myAtoi(char *str)
{
    if (*str == NULL)
       return 0;

    int res = 0;  // Initialize result
    int sign = 1;  // Initialize sign as positive
    int i = 0;  // Initialize index of first digit

    // If number is negative, then update sign
    if (str[0] == '-')
    {
        sign = -1;
        i++;  // Also update index of first digit
    }

    // Iterate through all digits of input string and update result
    for (; str[i] != '\0'; ++i)
    {
        if (isNumericChar(str[i]) == false)
            return 0; // You may add some lines to write error message
                      // to error stream
        res = res*10 + str[i] - '0';
    }

    // Return result with sign
    return sign*res;
}

// Driver program to test above function
int main()
{
    char str[] = "-134";
    int val = myAtoi(str);
```

```
    printf("%d ", val);
    return 0;
}
```

Time Complexity: O(n) where n is the number of characters in input string.

Exercise
Write your won atof() that takes a string (which represents an floating point value) as an argument and returns its value as double.

This article is compiled by **Abhay Rathi**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.
If you like GeeksforGeeks and would like to contribute, you can also write an article and mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Related Tpoics:

- Printing Longest Common Subsequence
- Suffix Array | Set 2 (nLogn Algorithm)
- Rearrange a string so that all same characters become d distance away

Recent Comments

affiszerv Your example has two 4s on row 3, that's why it...
Backtracking | Set 7 (Sudoku) · 13 minutes ago

RVM Can someone please elaborate this Qs from above...
Flipkart Interview | Set 6 · 33 minutes ago

Vishal Gupta I talked about as an Interviewer in general,...
Software Engineering Lab, Samsung Interview | Set 2 · 33 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

- 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)

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

34 Comments GeeksforGeeks

Sort by Newest

jinqueshu • 7 days ago

I think atoi() should be able to handle the following cases.

jinqueshu jinqueshu • 7 days ago

I think atoi() should be able to handle the following cases.

1. check NULL
-- You did, but it's wrong. should check if (str == NULL) instead if (*str

2. skip leading space
-- You didn't

3. handle negative sign '-'
-- You did a good job.

4. handle optional positive sign '+'
-- You didn't

5. handle non-digit character
-- You did. but usually should return the integer of valid part.

6. handle overflow
-- You didn't

7. handle underflow
-- You didn't

Reply • Share ›

Nikhil • 3 months ago

Sorry, but shouldn't we be checking for str==NULL before we de-reference it a

if (*str == NULL)
return 0;
add a check:
if(str==NULL)
// error processing ... exit(1) ?

Reply • Share ›

Pushkar Nikhil • 17 days ago

Yes... we should check the str pointer for NULL before checking wheth
as value or not.

Reply • Share ›

Coder011 • 3 months ago

atoi: http://ideone.com/gpjDSp
atof: http://ideone.com/O1HlIY

Recursive implementation of atoi and atof functions in C++.

Reply • Share ›

Chandra Mohan Sharma • 5 months ago

write your own atof function

```
#include<stdio.h>

#include<string.h>

void atof(char* );

void atof(char* num)

{

double sum = 0;

/* Calculate the integer part. */

while (*num)

{

if (*num!='.')
```

see more

1 ^ | v • Reply • Share ›

keye • 5 months ago

```
int my_atoi(char *str, int &k)

{

if(str == NULL)

{

cout<<"error, input string is NULL"<<endl; return="" -1;="" }="" int="" minus="0
```

)= {= minus= 1; p++;= }= while(p!= \0)= {= k= 10 (k) *= (p-'0
k="0" -="" (k);="" }="" return="" 1;="" }="">

sumit007 • 7 months ago

your function doesn't handle Overflow conditions.
Forex:
Input: "98784568924545654"
Output: String can't be expressed as valid number.

You can handle it by little modification in your final code before computing
res = res*10 + (str[i]-'0');

Modified Statement:
if(res > (INT_MAX - (str[i]-'0')) / 10) {
printf("%s\n", "Overflow !!! String can't be expressed as a valid number");
return 0;
}
else
res = res*10 + (str[i]-'0');

devC# • 8 months ago

public static int atoi(string str)

{

if (str == null || str.Length == 0)

throw new ArgumentNullException();

int value = 0;

```
for (var idx = str.Length - 1; idx >= 0; --idx)

{

if (idx == 0 && str[idx] == '-')

{

value = value * -1;

}
```

see more

• Reply • Share ›

minhaj • 8 months ago

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

int myatoi(char* str)
{ int output = 0; int i=0,sign=1;
if (*str=='-')
{sign=-1;
i++;
}
while( '0' <= str[i] && str[i] <='9' )
{ output = output*10 + (str[i] - '0') ;
i++;
}
return output*sign;
}

int main ()
{ int i; char buffer[256];
```

```
printf ( Enter a number: );
fgets (buffer, 256, stdin);
i = myatoi (buffer);
printf ("The value entered is %d. Its double is %d.\n",i,i*2);
return 0;
}
```

• Reply • Share ›

Arun Kumar • 9 months ago

Your program does not handle the test case of " 1234sdgds" in this case your

• Reply • Share ›

arun • 9 months ago

```
 /* Paste your code here (You may delete these lines if not writing c
/* atoi example */
#include <stdio.h>      /* printf, fgets */
#include <stdlib.h>     /* atoi */

int myatoi(char* str)
{
        int output = 0;
        int i=0;
        while( 48 <= str[i] && str[i] <=57 ){

                output = output*10 + (str[i] - 48) ;
                i++;
        }

        return output;
}
```

see more

• Reply • Share ›

Karthikeya Yakkali • 9 months ago

In the above code if(isNumericChar(str[i]) == false) return res instead of returr requirement

• Reply • Share ›

Kartikey • 9 months ago

why is -'0' appended at the end please explain //

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

1 • Reply • Share ›

Neo_T • 10 months ago

why is -'0' added in code for calculation of res? I am missing something.

• Reply • Share ›

Kartikey ➚ Neo_T • 9 months ago

k holds an integer, and it is recalculated by looking at the each charact you the character's ASCII code and '0? as well. The difference gives th shifting of previous value.

For example, if p is given by "425?,

The first calculation of k = 0*10 + (*p) – '0? = 0 + 52-48 = 4
The next calculation of k = 4*10 + (*p) – '0? = 40 + 50 – 48 = 42
The final calculation of k = 42*10 + (*p) – '0? = 420 + 53-48 = 425

Character ASCII code of "4? is 52, "2? is 50, "5? is 53, and "0? is 48.

Hope this would help you,

Ashish Kr Singh • 10 months ago

#include <stdio.h>
#include <ctype.h>
// A utility function to check whether x is numeric.
bool isNumericChar(char x).
{

return (x >= &#0390&#039 && x <= &#0399&#039)? true: false;
}

// A simple atoi() function. If the given string contains.
// any invalid character, then this function returns 0.
int myAtoi(char *str).
{

if (*str == NULL).

return 0;.

int res = 0; // Initialize result.

see more

• Reply • Share ›

Ashish Kr Singh • 10 months ago

#include <stdio.h>
#include <ctype.h>
// A utility function to check whether x is numeric.
bool isNumericChar(char x).
{

return (x >= &#0390&#039 && x <= &#0399&#039)? true: false;

}

// A simple atoi() function. If the given string contains.
// any invalid character, then this function returns 0.
int myAtoi(char *str).
{

if (*str == NULL).

return 0;.

int res = 0; // Initialize result.

see more

• Reply • Share ›

caijinlong • a year ago

```
 #include <iostream>
#include <string>
using namespace std;

int Atoi(string str)
{
        if (str.length() == 0)
                return 0;
        int result = 0;
        bool negative = false;
        int begin = 0;

        if (str[0] == '-')
        {
                begin = 1;
```

```
        }
```

see more

Reply • Share ›

caijinlong • a year ago

```
#include <iostream>

using namespace std;

int Atoi(char *str)
{
        if (str == NULL)
                return 0;
        int result = 0;
        bool negative = false;
        int begin = 0;

        if (str[0] == '-')
        {
                begin = 1;
                negative = true;
        }
```

see more

Reply • Share ›

Poorvank Bhatia • a year ago

When we input a string like "987hsg0987" to the atoi function it prints the soluti
code deals with this possibility!

1 ^ | v • Reply • Share ›

Tushar Patil • a year ago

```
int myAtoi(char *str)
{
int i=0;

while(*str)
{
i = (i << 3) + (i << 1) + (*str - '0');
str++;
}
return(i);
}
```

1 ^ | v • Reply • Share ›

Suhas Tilekar ➦ Tushar Patil • a year ago

Tussshhhh..
Nice one dude...!
think also for if entered string has -ve sign.

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

^ | v • Reply • Share ›

CODER_1 ➦ Tushar Patil • a year ago

Tussshhhh...

Nice .....!

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

Abhay ➚ Tushar Patil • a year ago

Good tricky implementation, love that :)

GeeksforGeeks • a year ago

Error checks have been added to the original code.

depak ➚ GeeksforGeeks • a year ago

you didn't check when string is very long. which will lead to buffer overf

depak ➚ depak • a year ago

```
cutoff = INT_MAX / 10;
cutlim = INT_MAX % 10;

c = str[i] - '0';
if (res > cutoff || (res == cutoff && c > cutlim))
overflow = 1;
else
{
res *= 10;
res += c;
}
```


Abhay ➚ depak • a year ago

My aim was to keep the post simple and readable, that is why c

Does standard atoi() do buffer overflow check, I don't think so. I

Let me know if u have a simple and readable code with buffer c

Reply • Share ›

GeeksQuiz • a year ago

Ahmed, Thanks for pointing this out. We have added code with error handling

Reply • Share ›

depak • a year ago

No null string check, no buffer overflow check !!!

Reply • Share ›

galaxy01 • a year ago

does it matter if you do for (;;++i) or for(;;i++)

Reply • Share ›

kartik galaxy01 • a year ago

As far as I know, compilers take care care of these optimizations and g
We don't need to worry about them.

Reply • Share ›

Ahmed Saleh • a year ago

There is a problem in your code. You must check also for values from 0 to 9 a
don&#039t also check for your inputs, you have to defense for a NULL input st

Reply • Share ›

Subscribe Add Disqus to your site