

← → ↺

leetcode.com/problems/string-to-integer-atoi/description/

☆ 9

🏠 Problem List < > 🔍

🏠 Run 🔄 Submit ⌚ 📄

88 ⚙️ 🔒 0 🧑 Premium

Description Editorial Solutions Submissions

< >

Code

8. String to Integer (atoi)

Medium Topics Companies

Implement the `myAtoi(string s)` function, which converts a string to a 32-bit signed integer.

The algorithm for `myAtoi(string s)` is as follows:

- Whitespace:** Ignore any leading whitespace (" ").
- Signedness:** Determine the sign by checking if the next character is '-' or '+', assuming positivity if neither present.
- Conversion:** Read the integer by skipping leading zeros until a non-digit character is encountered or the end of the string is reached. If no digits were read, then the result is 0.
- Rounding:** If the integer is out of the 32-bit signed integer range  $[-2^{31}, 2^{31} - 1]$ , then round the integer to remain in the range. Specifically, integers less than  $-2^{31}$  should be rounded to  $-2^{31}$ , and integers greater than  $2^{31} - 1$  should be rounded to  $2^{31} - 1$ .

Return the integer as the final result.

**Example 1:**

Input: s = "42"

Output: 42

Java Auto

```
1 class Solution {
2     public int myAtoi(String s) {
3         s = s.trim(); // Remove leading whitespace
4         int sign = 1, i = 0;
5         long res = 0; // Using long to handle overflow cases
6
7         if (s.length() == 0) return 0;
8
9         // Check for sign
10        if (s.charAt(0) == '-') { sign = -1; i++; }
11        else if (s.charAt(0) == '+') { i++; }
12
13        // Process numerical characters
14        while (i < s.length()) {
15            char ch = s.charAt(i);
16            if (ch < '0' || ch > '9') break; // Stop at non-numeric character
17        }
```

Saved

Ln 26, Col 1

Testcase Test Result

Accepted Runtime: 0 ms

• Case 1 • Case 2 • Case 3 • Case 4 • Case 5

Input

s =

"42"

5.2K 695 163 Online