

剑指 Offer 67. 把字符串转换成整数

Monday, June 27, 2022 9:32 AM

<https://leetcode.cn/problems/ba-zi-fu-chuan-zhuan-huan-cheng-zheng-shu-lcof/>
中等

题目简而言之就是，将一个字符串转为整数，数字前面有空格可以、正负号可以，但是不能有其他字母，遍历数字到有字母、空格或结尾结束。转不了返回0，大于小于最大值最小值返回最大值最小值。

首先，该函数会根据需要丢弃无用的开头空格字符，直到寻找到第一个非空格的字符为止。当我们寻找到的第一个非空字符为正或者负号时，则将该符号与之后面尽可能多的连续数字组合起来，作为该整数的正负号；假如第一个非空字符是数字，则直接将其与之后连续的数字字符组合起来，形成整数。

该字符串除了有效的整数部分之后也可能会存在多余的字符，这些字符可以被忽略，它们对于函数不应该造成影响。

注意：假如该字符串中的第一个非空格字符不是一个有效整数字符、字符串为空或字符串仅包含空白字符时，则你的函数不需要进行转换。

在任何情况下，若函数不能进行有效的转换时，请返回 0。

说明：

假设我们的环境只能存储 32 位大小的有符号整数，那么其数值范围为 $[-2^{31}, 2^{31} - 1]$ 。如果数值超过这个范围，请返回 `INT_MAX (2^{31} - 1)` 或 `INT_MIN (-2^{31})`。

解法：

先去除首尾空格存入字符数组。

定义符号位标记首位的符号，定义数字的起始位标记，默认位1，正数无符号则从0开始遍历。

定义最大值为最大正数除10，因为要对最后一位额外判断。

```
class Solution {
    public int strToInt(String str) {
        // 去除首尾空格
        char[] c = str.trim().toCharArray();
        if(c.length == 0) return 0;
        int res = 0, bndry = Integer.MAX_VALUE / 10; // 最大正数为2147483647，需要对最后一位额外判断，故最大正整数先除以10
        int i = 1, sign = 1; // sign记录符号位，默认标记正号 1； i作为默认遍历起点
        if(c[0] == '-') sign = -1; // 首位为负号，故符号位为-1
        else if(c[0] != '+') i = 0; // 首位不是负号和正号时，循环从首位0开始
        for(int j = i; j < c.length; j++) {
            if(c[j] < '0' || c[j] > '9') break; // 若遇到不为数字的位 跳出循环
            // 若结果大于最大值 或 结果等于最大值且末位大于7 则越界（此时最大值为真正的最大值除以10的结果，故当前值若是已经大于bndry最后一位不用判断了）
            if(res > bndry || res == bndry && c[j] > '7')
```

```
        return sign == 1 ? Integer.MAX_VALUE : Integer.MIN_VALUE; //越界则根据
        标记位返回最大值或最小值
        res = res * 10 + (c[j] - '0'); // 正常遍历 记录此时的res值
    }
    return sign * res; // 正常结束 返回res值。
}
}
```