Description | Solution | Discuss (999+) | Submissions

Java | Autocomplete

394. Decode String

Medium 3843 188 Add to List Share

Given an encoded string, return its decoded string.

The encoding rule is: `k[encoded_string]`, where the *encoded_string* inside the square brackets is being repeated exactly *k* times. Note that *k* is guaranteed to be a positive integer.

You may assume that the input string is always valid; No extra white spaces, square brackets are well-formed, etc.

Furthermore, you may assume that the original data does not contain any digits and that digits are only for those repeat numbers, *k*. For example, there won't be input like `3a` or `2[4]`.

Example 1:

```
Input: s = "3[a]2[bc]"
Output: "aaabcbc"
```

Example 2:

```
Input: s = "3[a2[c]]"
Output: "accaccacc"
```

Example 3:

```
Input: s = "2[abc]3[cd]ef"
Output: "abcabccdcdcdef"
```

Example 4:

```java
class Solution {
    public String decodeString(String s) {
        Stack<Integer> intStack = new Stack<>();
        Stack<StringBuilder> strStack = new Stack<>();
        StringBuilder cur = new StringBuilder();
        int k = 0;
        for (char ch : s.toCharArray()) {
            if (Character.isDigit(ch)) {
                k = k * 10 + ch - '0';
            } else if ( ch == '[') {
                intStack.push(k);
                strStack.push(cur);
                cur = new StringBuilder();
                k = 0;
            } else if (ch == ']') {
                StringBuilder tmp = cur;
                cur = strStack.pop();
                for (k = intStack.pop(); k > 0; --k) cur.append(tmp);
            } else cur.append(ch);
        }
        return cur.toString();
    }
}
```

Testcase | Run Code Result | Debugger

Accepted Runtime: 0 ms

Your input `"3[a]2[bc]"`

Output `"aaabcbc"` Diff

Expected `"aaabcbc"`