

# 394. Decode String

结果"3[a2[c]]" 2c展开以后重新push进stack

```
public String decodeString(String s) {
    Deque<Integer> num = new ArrayDeque<>();
    Deque<Character> words = new ArrayDeque<>();
    int i = 0, start = 0;
    StringBuilder sb = new StringBuilder();
    StringBuilder res = new StringBuilder();
    while(i < s.length()){
        if(isNumber(s.charAt(i))){
            start = i;
            while(i < s.length() && isNumber(s.charAt(i)))
                i++;
            num.push(Integer.parseInt(s.substring(start, i)));
        }else if(s.charAt(i) == ' '){
            while(!words.isEmpty() && words.peek() - '[' != 0){
                sb.append(words.pop());
            }
            words.pop();
            int k = num.pop();
            sb.reverse();
            String tmp = sb.toString();
            for(int j = 0; j < k; ++j){
                for(int m = 0; m < tmp.length(); ++m){
                    words.push(tmp.charAt(m));
                }
            }
            sb.setLength(0);
            i++;
        }else{
            words.push(s.charAt(i));
            i++;
        }
    }
    for(char c : words){
        res.append(c);
    }
    return res.reverse().toString();
}

public boolean isNumber(char c){
    if(c - '0' >= 0 && c - '9' <= 0)
        return true;
    return false;
}
```

偷的别人的

```
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();

```