

兄弟们，英语解释看累了没，我来个中文版的~

493

VIEWS

思路：

- 建立一个栈`stk[]`，`top = 0`
- 设一个变量`digits`，`digits = len(nums) - k`表示最后答案的长度（仔细体会`digits`的妙用）
- `for`从`nums[0]`开始循环扫描到末尾（或`0 == k`时）
 - `while`栈顶元素`stk[top - 1] > num[i]`（大概的意思就是把较高位中比较大的数移除），`pop()`栈顶元素，`top--`，`k--`（需要移除个数减一）（一直`while`直到条件不满足）
 - 每次`for`无论什么情况，最后都`push(num[i])`，`top++`（栈空了当然就直接`push`）
- 最后从`stk[0]`开始取前`digits`个元素，即为答案（若有前导`0`，则从非`0`开始取）

贴一下最高票的答案供参考：

```
public class Solution {
    public String removeKdigits(String num, int k) {
        int digits = num.length() - k;
        char[] stk = new char[num.length()];
        int top = 0;
        // k keeps track of how many characters we can remove
        // if the previous character in stk is larger than the current one
        // then removing it will get a smaller number
        // but we can only do so when k is larger than 0
        for (int i = 0; i < num.length(); ++i) {
            char c = num.charAt(i);
            while (top > 0 && stk[top-1] > c && k > 0) {
                top -= 1;
                k -= 1;
            }
            stk[top++] = c;
        }
    }
}
```

```
        stk[top++] = c;
    }
    // find the index of first non-zero digit
    int idx = 0;
    while (idx < digits && stk[idx] == '0') idx++;
    return idx == digits? "0": new String(stk, idx, digits - idx);
}
}
```

特殊情况解释

有人指出 `nums = "123"`，`k = 2` 时的情况怎么办？

这时，该算法判断语句没用上，一直在 `push`，则 `stk` 就是 `123` 不是 `1`？

可别忘了还有 `digits`，我们要取的就是前 `digits` 个元素，所以最后答案就是 `1`。

老弟是个新手，所以感觉此法甚妙，不知各位前辈有何感想。

要是描述不明白的地方，还望担待。