

Problem 1881: Maximum Value after Insertion

Problem Information

Difficulty: Medium

Acceptance Rate: 38.96%

Paid Only: No

Tags: String, Greedy

Problem Description

You are given a very large integer `n`, represented as a string,  and an integer digit `x`. The digits in `n` and the digit `x` are in the **inclusive** range `[1, 9]`, and `n` may represent a **negative** number.

You want to **maximize** `n`'s numerical value** by inserting `x` anywhere in the decimal representation of `n` . You **cannot** insert `x` to the left of the negative sign.

* For example, if `n = 73` and `x = 6`, it would be best to insert it between `7` and `3`, making `n = 763`. * If `n = -55` and `x = 2`, it would be best to insert it before the first `5`, making `n = -255`.

Return _a string representing the**maximum** value of _`n`  after the insertion_.

Example 1:

Input: n = "99", x = 9 **Output:** "999" **Explanation:** The result is the same regardless of where you insert 9.

Example 2:

Input: n = "-13", x = 2 **Output:** "-123" **Explanation:** You can make n one of {-213, -123, -132}, and the largest of those three is -123.

Constraints:

* `1 <= n.length <= 105` * `1 <= x <= 9` * The digits in `n` are in the range `[1, 9]`. * `n` is a valid representation of an integer. * In the case of a negative `n`, it will begin with `-`.

Code Snippets

C++:

```
class Solution {  
public:  
    string maxValue(string n, int x) {  
  
    }  
};
```

Java:

```
class Solution {  
public String maxValue(String n, int x) {  
  
}  
}
```

Python3:

```
class Solution:  
    def maxValue(self, n: str, x: int) -> str:
```