LeetCode Multiply Strings: Simulation of Add and Multiplication

Overview

LeetCode Multiply Strings: the challeng is that the value could be arbitrary large, so we have to simulate Add and Multiplication character by character.

LeetCode Multiply Strings

Given two numbers represented as strings, return multiplication of the numbers as a string.

Note: The numbers can be arbitrarily large and are non-negative.

Solution and Precautions: Simulation of Add and Multiplication

The algorithm is not difficult, just a simulation of the way we do multiplication, we need exact each single character out and do the multiplication on step by one step. Just be careful when dealing with the carry and the zero result. Several special cases worth noting:

999 * 0 = 0

501 * 6 = 3006

The following Java code is accepted by LeetCode OJ to pass this Multiply Strings problem:

```
class Solution {

    private String mul(String s, char c, int j) {
        if ('0' == c)
            return "";
```

```
StringBuffer sb = new StringBuffer();
while (j-- > 0)
    sb.append(0);

int mul = Character.getNumericValue(c);
int carry = 0;
int sum = 0;

boolean isPositive = false;

for (int i = s.length() - 1; i >= 0; --i) {
    sum = carry + Character.getNumericValue(s.charAt(i)) * mul;
    carry = sum / 10;
    sum %= 10;
    if (sum > 0)
        isPositive = true;
    sb.append(sum);
}

if (carry > 0) {
    sb.append(carry);
    isPositive = true;
```

```
        }

        if (!isPositive)
            return "";

        return sb.reverse().toString();
    }

    private String add(String a, String b) {
        StringBuffer sb = new StringBuffer();
        int carry = 0;
        int sum = 0;

        int i = a.length() - 1;
        int j = b.length() - 1;

        int aa = 0;
        int bb = 0;

        while (i >= 0 || j >= 0) {
            aa = i >= 0 ? Character.getNumericValue(a.charAt(i--)) : 0;
            bb = j >= 0 ? Character.getNumericValue(b.charAt(j--)) : 0;
            sum = carry + aa + bb;
            carry = sum / 10;
            sum %= 10;
            sb.append(sum);
        }
        if (carry > 0)
            sb.append(carry);

        return sb.reverse().toString();
    }

    public String multiply(String num1, String num2) {
        String ret = "";
        for (int i = num2.length() - 1, f = 1; i >= 0; --i, f *= 10) {
            ret = add(ret, mul(num1, num2.charAt(i), num2.length() - i - 1));
        }
        if (ret.length() == 0)
            return "0";
        return ret;
    }
}
```

Summary

LeetCode Multiply Strings: the challeng is that the value could be arbitrary large, so we have to simulate Add and Multiplication character by character.

Written on May 20, 2013

« LeetCode Maximum Subarray: 4 methods, DP, Divide and Conquer

LeetCode Partition List: Two Pointers and O(1) Space »