43. Multiply Strings

Tags	medium
Property	@October 18, 2022
URL	

Question

原文：

Given two non-negative integers `num1` and `num2` represented as strings, return the product of `num1` and `num2`, also represented as a string.

Note: You must not use any built-in BigInteger library or convert the inputs to integer directly.

我的理解：

給兩個大數進行乘法，但大數是由字串的形式存在不可以直接轉成數字型別進行計算。

翻譯：

自評翻譯正確性：100

- Word Memory：

Code

```
class Solution {
public:
    string multiply(string num1, string num2) {
        int i=0,j=0,z=0;
        int add=0,temp=0,advancement=0;
        //tempste當前num2數字乘以num1的結果
        //ansstr當前總和
        //ANS準備承接當前總和+新算好的計算結果=所有總和
        string tempstr="",ansstr="",ANS="";

        if(num1=="0"||num2=="0")
            return "0";

        for(i=num2.size()-1;i>=0;i--){
            tempstr="";
            add=0;
```

```
            advancement=0;
            for(j = num1.size()-1;j >= 0;j--){//做乘法
                temp = 0;
                temp += advancement+(num1[j]-'0') * (num2[i]-'0');
                tempstr = to_string(temp % 10) + tempstr;
                advancement = temp / 10;
            }
            if(advancement!=0){//如果有進位到溢位也要補在前頭
                tempstr=to_string(advancement)+tempstr;
            }

            //第二行計算結果開始補0 方便等等作加法
            //  ****
            // ****0
            //****00
            z=i;
            while(z<num2.size()-1){
                tempstr+="0";
                z++;
            }

            while(ansstr.size()<tempstr.size()){//總和長度至少要等於當前計算結果的長度方便做字串處理
                ansstr="0"+ansstr;
            }

            advancement=0;//進位變數沿用
            ANS="";//ANS用來接 總和+當前計算結果 的結果
            add=ansstr.size()-1;
            while(add>=0){//做加法
                temp = 0;
                temp +=advancement+ (ansstr[add]-'0')+(tempstr[add]-'0');
                ANS = to_string(temp % 10) + ANS;
                advancement = temp / 10;
                add--;
            }

            if(advancement!=0){//如果有進位到溢位也要補在前頭
                ANS=to_string(advancement)+ANS;
            }
            ansstr=ANS;//目前所有總和結果
        }
        return ANS;
    }
};
```

> 思路：
>
> - 先從num2的個位數依序乘以整個num1，並記錄計算結果tenpstr
> - 這個計算結果再去加上目前總和ansstr得到所有總和ANS

Success Details >

Runtime: 440 ms, faster than 5.01% of C++ online submissions for Multiply Strings.

Memory Usage: 63.1 MB, less than 5.32% of C++ online submissions for Multiply Strings.

Next challenges:

Add Two Numbers | Plus One | Add Binary | Add Strings | Apply Discount to Prices

Show off your acceptance:

Time Submitted	Status	Runtime	Memory	Language
10/18/2022 15:27	Accepted	440 ms	63.1 MB	cpp

優良code参考

```
class Solution {
public:
    string multiply(string num1, string num2) {
        if (num1 == "0" || num2 == "0") return "0";

        vector<int> res(num1.size()+num2.size(), 0);

        for (int i = num1.size()-1; i >= 0; i--) {
            for (int j = num2.size()-1; j >= 0; j--) {
                res[i + j + 1] += (num1[i]-'0') * (num2[j]-'0');
                res[i + j] += res[i + j + 1] / 10;
                res[i + j + 1] %= 10;
            }
        }

        int i = 0;
        string ans = "";
        while (res[i] == 0) i++;
        while (i < res.size()) ans += to_string(res[i++]);
```

```
        return ans;
    }
};
```

> 思路：
>
> - 先宣告陣列 res 用來承接計算結果（最後計算結果的長度最多就是跟num1+num2的長度相同，所以可以先宣告 res 大小）
> - for迴圈就是直接在做乘法的計算，因為都有固定的位置不用特別處理溢位或結果長度的問題
> - 最後在把這個陣列的結果輸出成字串