题目：

Given a string of numbers and operators, return all possible results from computing all the different possible ways to group numbers and operators. The valid operators are +, - and *.

Example 1

Input: "2-1-1".

```
((2-1)-1) = 0

(2-(1-1)) = 2
```

Output: [0, 2]

Example 2

Input: "2*3-4*5"

```
(2*(3-(4*5))) = -34

((2*3)-(4*5)) = -14

((2*(3-4))*5) = -10

(2*((3-4)*5)) = -10

(((2*3)-4)*5) = 10
```

Output: [-34, -14, -10, -10, 10]

1.时间：O（N^2）；空间：O（N）

```
class Solution {
public:
    vector<int> diffWaysToCompute(string input) {
        std::vector<int> result;
        for (int i = 0; i < input.size(); ++i){
            if (input[i] != '+' && input[i] != '-' && input[i] != '*') continue;
            std::vector<int> left = diffWaysToCompute(input.substr(0, i));
            std::vector<int> right = diffWaysToCompute(input.substr(i + 1));
            for (int j = 0; j < left.size(); ++j){
                for (int k = 0; k < right.size(); ++k){
                    if (input[i] == '+') result.push_back(left[j] + right[k]);
                    else if (input[i] == '-') result.push_back(left[j] - right[k]);
                    else if (input[i] == '*') result.push_back(left[j] * right[k]);
                }
            }
        }
        /* 若无运算符，则表明仅有数字，直接插入到结果中 */
        if (result.empty()) result.push_back(atoi(input.c_str()));
        return result;
    }};
```