给定一个含有数字和运算符的字符串，为表达式添加括号，改变其运算优先级以求出不同的结果。你需要给出所有可能的组合的结果。有效的运算符号包含 +, - 以及 * 。

解法:

```
class Solution {
public:
  std::vector<int> diffWaysToCompute(std::string expression) {
    std::vector<int> res;
    for (int i = 0; i < expression.size(); i++) {
      if (expression[i] == '+' || expression[i] == '*' ||
          expression[i] == '-') {
        // 分治
        std::vector<int> left  = diffWaysToCompute(expression.substr(0,
i));
        std::vector<int> right = diffWaysToCompute(
            expression.substr(i + 1, expression.size() - i - 1));

        for (auto item : left) {
          for (auto it : right) {
            switch (expression[i]) {
              case '+':
                res.push_back(item + it);
                break;
              case '-':
                res.push_back(item - it);
                break;
              case '*':
                res.push_back(item * it);
                break;
            }
          }
        }
      }
    }

    if (res.empty()) {
      res.push_back(atoi(expression.c_str()));
    }

    return res;
  }
};
```