题目：

Given two integers representing the numerator and denominator of a fraction, return the fraction in string format.

If the fractional part is repeating, enclose the repeating part in parentheses.

For example,

- **Given numerator = 1, denominator = 2, return "0.5".**
- **Given numerator = 2, denominator = 1, return "2".**
- **Given numerator = 2, denominator = 3, return "0.(6)".**

```
class Solution {
    typedef long long int64_t;
public:
    string fractionToDecimal(int numerator, int denominator) {
        if (numerator == 0) return "0";

        std::string result = ((numerator >> 31) ^ (denominator >> 31)) ? "-" : "";
        int64_t _numerator = numerator;
        int64_t _denominator = denominator;

        int64_t _numerator = std::abs(_numerator);     /*  std::abs(-2147483648)  =
-2147483648 */
        int64_t _denominator = std::abs(_denominator);

        result += std::to_string(_numerator / _denominator);
        if (_numerator % _denominator == 0)/* 如果没有小数部分，那么直接返回 */
            return result;

        result += ".";
        int64_t remainder = _numerator % _denominator;

        std::unordered_map<int, int> hashTable;
        while (remainder != 0 && hashTable.find(remainder) == hashTable.end()){
            hashTable.insert(std::pair<int, int>(remainder, result.size()));     /* 保存
result.size()是为了找到循环体的开始位置 */
            remainder *= 10;
            result += std::to_string(remainder / _denominator);
            remainder %= _denominator;
        }

        if (remainder != 0){
            result.insert(hashTable[remainder], 1, '(');
            result += ")";
        }

        return result;
    }
};
```