# 题目

给定两个整数，分别表示分数的分子 numerator 和分母 denominator，以 字符串形式返回小数 。

如果小数部分为循环小数，则将循环的部分括在括号内。

如果存在多个答案，只需返回 任意一个 。

对于所有给定的输入，保证 答案字符串的长度小于 104 。

示例 1：

输入：numerator = 1, denominator = 2

输出："0.5"

示例 2：

输入：numerator = 2, denominator = 1

输出："2"

示例 3：

输入：numerator = 4, denominator = 333

输出："0.(012)"

提示：

-231 <= numerator, denominator <= 231 - 1

denominator != 0

# 分析

要解决将分数转换为小数（含循环小数处理）的问题，需处理整数部分、小数部分及循环节的识别。核心在于通过记录余数出现的位置，判断是否存在循环节。

解题思路

1、处理符号：若分子和分母异号，结果需带负号。先确定符号，再将分子和分母转为正数处理（避免负数运算的复杂性）。

2、整数部分计算：通过除法得到商的整数部分，剩余的余数用于计算小数部分。

3、小数部分与循环节识别：

- 若余数为0，说明是有限小数，直接返回结果。

- 若余数非0，进入小数部分计算：每次将余数乘10，除以分母得到当前小数位，更新余数。

- 用哈希表记录每个余数首次出现的位置：若余数再次出现，说明从首次位置到当前位置的小数部分是循环节，需用括号括起。

4、拼接结果：组合整数部分、小数点（若有小数）、非循环小数部分和循环节（若存在）。

代码：

class Solution {

public:

string fractionToDecimal(int numerator, int denominator) {

if (numerator == 0) return "0"; // 特殊情况：分子为0

string result;

// 处理符号（异号则结果为负）

if ((numerator > 0) ^ (denominator > 0)) {

result += "-";

}

// 转为长整数避免溢出（如分子为INT\_MIN时取绝对值会溢出）

long long num = abs((long long)numerator);

long long den = abs((long long)denominator);

// 计算整数部分

result += to\_string(num / den);

long long remainder = num % den;

if (remainder == 0) {

return result; // 无小数部分

}

// 处理小数部分

result += ".";

unordered\_map<long long, int> remainder\_pos; // 记录余数首次出现的位置

while (remainder != 0) {

if (remainder\_pos.count(remainder)) {

// 余数重复出现，找到循环节

result.insert(remainder\_pos[remainder], "(");

result += ")";

break;

}

// 记录当前余数的位置

remainder\_pos[remainder] = result.size();

// 计算下一位小数

remainder \*= 10;

result += to\_string(remainder / den);

remainder %= den;

}

return result;

}

};

代码解释

1、符号处理：通过异或运算判断分子和分母是否异号，若异号则在结果开头添加负号。

2、整数部分：直接用分子除以分母得到整数商，若余数为0，直接返回整数部分。

3、小数部分与循环节：

- 余数非0时，添加小数点开始处理小数部分。

- 用哈希表remainder\_pos存储每个余数首次出现的位置（结果字符串的长度）。

- 每次将余数乘10，除以分母得到当前小数位，更新余数。若余数已在哈希表中，说明从首次位置到当前位置是循环节，插入括号后退出循环。

4、避免溢出：将分子和分母转为long long类型，防止处理INT\_MIN（如-2147483648）时取绝对值导致的溢出。

复杂度分析

- 时间复杂度：O(k)，其中k是结果字符串的长度（最多10^4，由题目约束保证）。每个余数最多出现一次，循环次数受限于此。

- 空间复杂度：O(k)，哈希表存储的余数数量不超过结果字符串的长度。