1. Fraction to Recurring Decimal

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.

**Example 1:**

Input: numerator = 1, denominator = 2  
Output: "0.5"

**Example 2:**

Input: numerator = 2, denominator = 1  
Output: "2"

**Example 3:**

Input: numerator = 2, denominator = 3  
Output: "0.(6)"

**解** 当余数重复出现时，意味着循环节出现了。因此需要用一个字典pos记录每个余数出现的位置，重复出现的余数对应的两个位置之间的即为循环节

Note: 运行时发现数字会出现溢出的问题，代码中的类型改成了long long int

class Solution {  
public:  
 string fractionToDecimal(long long int numerator, int denominator) {  
 string res;  
 if(numerator < 0 && denominator > 0){  
 numerator \*= -1;  
 res += "-";  
 }else if(numerator > 0 && denominator < 0){  
 denominator \*= -1;  
 res += "-";  
 }  
 res += to\_string(divmod(numerator, denominator));  
 if(numerator == 0)return res;  
 res += ".";  
 map<int, int>mp;  
 mp[numerator] = res.size();  
 while(numerator){  
 numerator \*= 10;  
 int tmp = divmod(numerator, denominator);  
 res += to\_string(tmp);  
 if(mp.find(numerator) != mp.end()){  
 res.insert(mp[numerator], "(");  
 res += ")";  
 break;  
 }  
 mp[numerator] = res.size();  
 }  
 return res;  
  
 }  
 long long int divmod(long long int &n1, int &n2){  
 long long int res = n1 / n2;  
 n1 -= res \* n2;  
 return res;  
 }  
};