Given a 32-bit signed integer, reverse digits of an integer.

**Example 1:**

Input: 123  
Output: 321

**Example 2:**

Input: -123  
Output: -321

**Example 3:**

Input: 120  
Output: 21

**Note:** Assume we are dealing with an environment which could only store integers within the 32-bit signed integer range: [−231, 231 − 1]. For the purpose of this problem, assume that your function returns 0 when the reversed integer overflows.

**注意处理溢出的问题，可以用字符串处理，也可以存为long long int结合进制转换**

class Solution {  
public:  
 int reverse(int x) {  
 long long int ans = 0, temp = x;  
 int sign = 1;  
 if(x < 0){  
 sign = -1;  
 temp = -temp;  
 }  
 do{  
 ans = ans \* 10 + temp % 10;  
 temp /= 10;  
 }while(temp != 0);  
 if(ans > INT\_MAX)return 0;  
 return ans \* sign;  
 }  
};