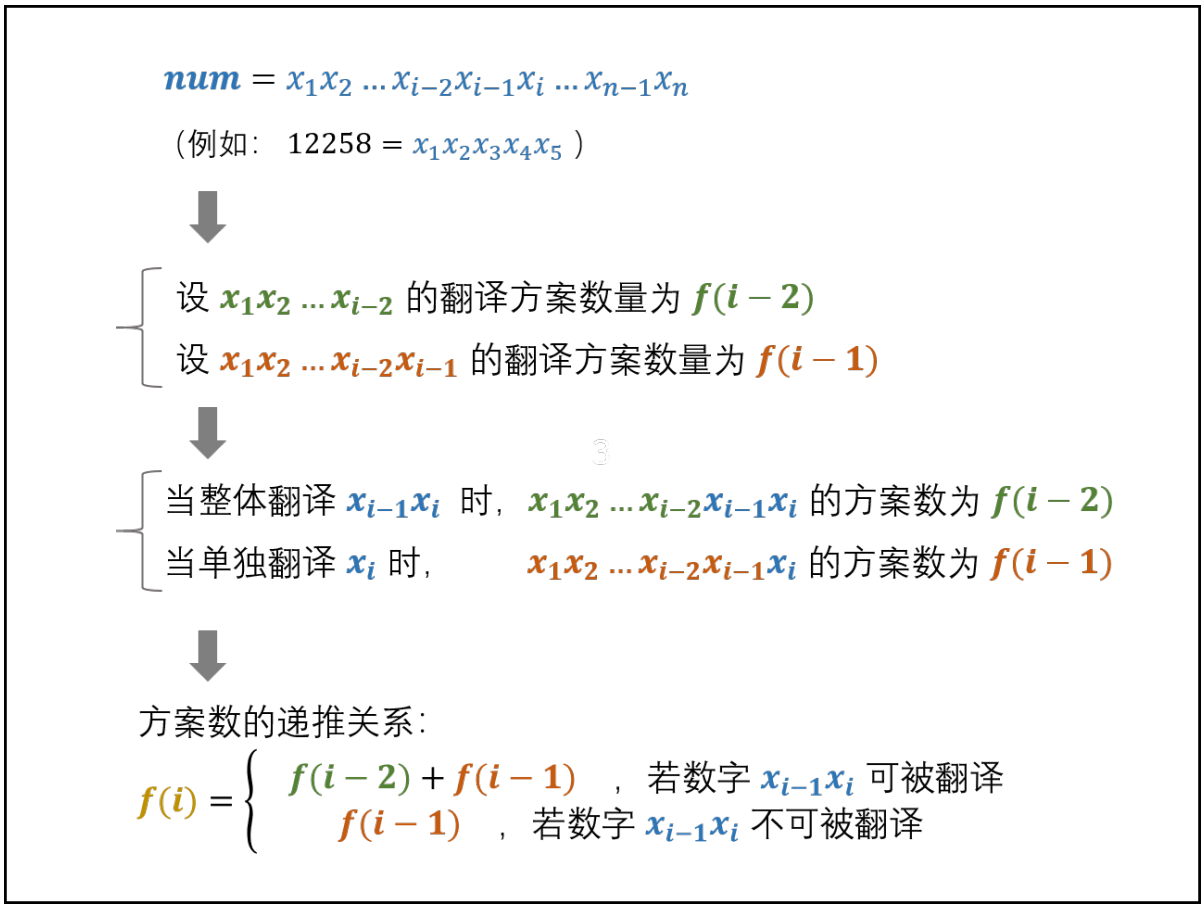
剑指46 把数字翻译成字符串

本质还是二维动态规划，依然是使用2个变量交替前进优化空间使用。

但是状态转移方程有不同情况，正向的话思考有点纠结，第一次做耗时很多。




```cpp
// my answer
class Solution {
public:
    int translateNum(int num) {
        int dpshort = 1, dpmid = 1, dplong = 1;
        while (num/10 != 0) {
            if (num % 100 <= 25 && num % 100 >= 10) {
                dplong = dpmid + dpshort;
            }
            else {
                dplong = dpmid;
            }
            dpshort = dpmid;
            dpmid = dplong;
            num /= 10;
        }
        return dplong;
        //从最后1个数字到倒数第n个数字。
```

```
        // dp[n] = dp[n-1] + dp[n-2]  (从包含倒数第n个数字向右的二位数 <=25 且 >=10 )
        // dp[n] = dp[n-1]   (二位数 > 25)
    }
};
```