

编程题-1

解题思路:

基本思路就是按照模板的思路来做，也就是模拟竖式加法。

我有三个自己的变量 len 有效位个数, maxlen 两个加数中最大的有效位个数, flag 标识最终结果的有效位个数是 maxlen 还是 maxlen+1.

```
private int flag = 0;
```

构造函数: 因为 non-Number type 是非法的, 所以要将输入的 string 转换成一个 int 数组, 为了方便最后的加法, 我使这个数组的前面全 0 有效位放在数组的最后部分。

add 函数: 模拟竖式加法的过程。

toString 函数: 因为结果的存储方式是 int 型数组, 所以要把它转成 string 类型同时去除为了方便进行加法而填充的 0。

编程题-2

解题思路:

基本思路是分别验证数独的行列和子矩阵是否符合要求。验证的方式是根据鸽巢原理, 在同一行一列一个子矩阵中, 一个数出现不能超过 2 次, 否则这个数独一定不正确, 便直接返回 false。

我建立了一个 Sudoku 类, 来分别实现对列行, 子矩阵的验证。Main 类只是作为程序的入口并实现数独的读取。

col_row 函数: 验证行和列, 我把这两个验证合并在了一起, 这样能减少复杂度的常数。

sub_matrix 函数: 验证子矩阵是否合法。