本次结对项目，我们选择的题目是数独解的生成。

数独是源自18世纪瑞士的一种数学游戏。是一种运用纸、笔进行演算的逻辑游戏。玩家需要根据9×9盘面上的已知数字，推理出所有剩余空格的数字，并满足每一行、每一列、每一个粗线宫（3\*3）内的数字均含1-9，不重复。数独盘面是个九宫，每一宫又分为九个小格。在这八十一格中给出一定的已知数字和解题条件，利用逻辑和推理，在其他的空格上填入1-9的数字。使1-9每个数字在每一行、每一列和每一宫中都只出现一次，所以又称“九宫格”。

利用程序去解决数独的填写问题，最直接的想法就是遍历然后检验逐次填入的数字是否满足数独的规范要求。如果不符合要求，则回溯到上一步的数字选择，重新选择一个之前没有选择的数字。所以设计起来很简单，整体上共分为两个部分：

第一部分是数字的填入，我们选择的方式是随机填入。因为如果让程序固定从1到9循环的填入的话，这样在程序中生成的数独就是完全一样的结构了，故结果只有九个不同的数独。所以我选择将数独空格中随机填入1-9的数字。第二部分就是验证数字规则的符合性，开始时我们写了一个复杂的函数来完成填数字和验证的全部操作。这样整个代码就显得比较臃肿，不是很简洁。所以我进行了代码重构，将验证部分抽取出来单独做成了一个函数。这样从代码上来说程序由两部分组成：填数字和验证数字。

函数IsSuitable(),有两个参数：row(int)和col(int),两者分别表示的是行和列，用于验证某行列中的数字是否满足条件。返回一个布尔类型的数据，true表示符合条件。而false表示不符合条件。整个函数的逻辑也很简单，就是检测同行、同列以及所在九宫格中是否已经存在相同的元素。函数generate()，有两个参数有两个参数row(int)和col(int),,两者分别表示的是行和列，函数从这里开始，遵循行优先的原则逐次填入数字。采用递归的方式，如果当前填的数字符合规则的要求，则判断当前是否填满，因为是逐行逐列填充的数字，所以我们只要检验右下角的空格（row=8,col=9）是否填充来判断整个的矩阵是否填充完毕。如果没有填充完毕，则依次检验行满否、列满否。当程序右下空格填满时，返回true，递归结束。如果尝试所有数字都不符合条件的话，循环结束，返回false，回溯到上一步重新计算。

此外，还有两个功能性的函数：outputTocmd()和outputTotext(),两者分别通过循环将生成好的数独解打印到终端中和本地磁盘系统中。

总结，程序整体有四个函数：

1. IsSuitable(int row, int col)
2. generate(int row, int col)
3. outputTocmd()
4. outputTotext()