我主要把项目分为三部分:

输入部分

生成数独终局

解数独

在写界面的时，只要稍作修改就可以使用后两部分。

详细描述一下这三个部分:

输入略过，有Python的异常处理很好写

生成数独终局用的是行列变换法。数独中间的九宫格经过行列变换可以变换为2!×3!×3!×2!×3!×3!=5184（因为固定左上角为5）。这样我需要生成1000000/5184=192个九宫格就够了，而正中九宫格有8!=40320个足够满足要求。

解数独我主要参考了《数独求解的候选数优化算法设计》这篇论文，运用了显性候选数规则、隐性候选数规则、九宫格交叉排除规则。Python的numpy可以很好的对数组进行计算。在进行dfs时，我对候选数进行了估值，具体是(10-候选个数) + 同行确定数字个数 + 同列确实数字个数

