**基于SAT的数独游戏求解程序”用户使用手册**

**内容提要**

**本手册包含以下三个方面的内容：**

* **程序主页面（菜单页面）介绍**
* **求解cnf文件中的SAT问题介绍**
* **数独游戏介绍**

**其中，程序主页面是用户编译程序之后看到的页面。用户可以根据交互提示选择近视求解cnf程序或者进入数独游戏程序，详情请见后面对于三个模块的具体操作使用方法的介绍**

* 1. **程序主页面（菜单页面）介绍**
* **编译源文件main.cpp。（因为设计效果的问题，有些地方需要用c++语法，除了个别地方，其他均为c语言语法）**
* **你会看到如下界面**

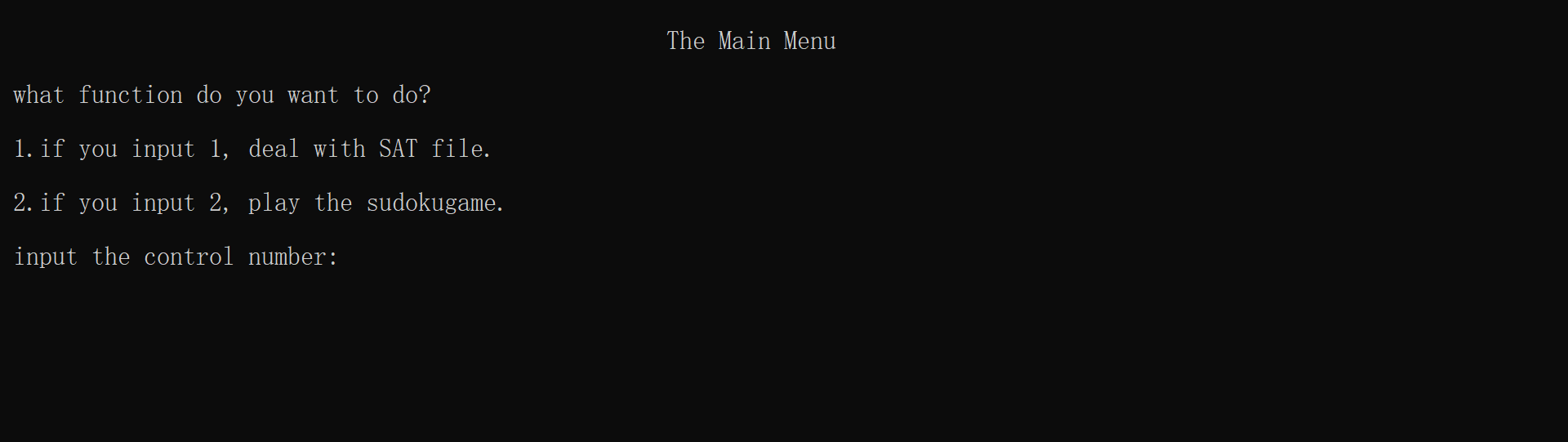


图1-1 主菜单示意图

**你需要做的是输入1或者2。**

* 其中输入1代表你将要用DPLL算法求解SAT问题（求解cnf文件中的SAT问题）
* 输入2代表你要玩数独游戏
* **比如输入1，你会看到下左图，回车后清屏，进入SAT程序，如下右图**

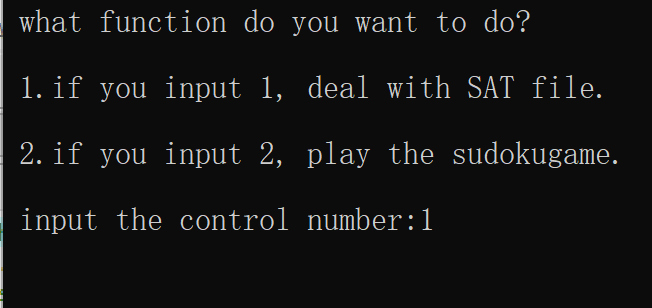
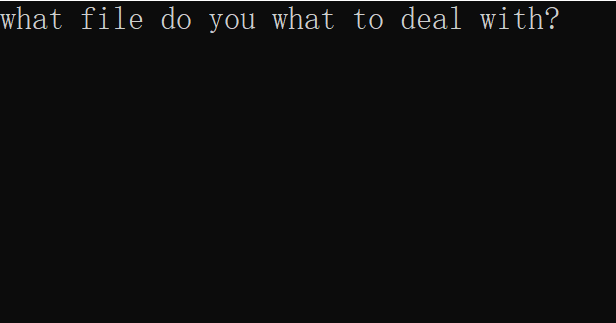
 

图1-2 输入1后界面示意图

* **输入2，回车后清屏，进入数独游戏**

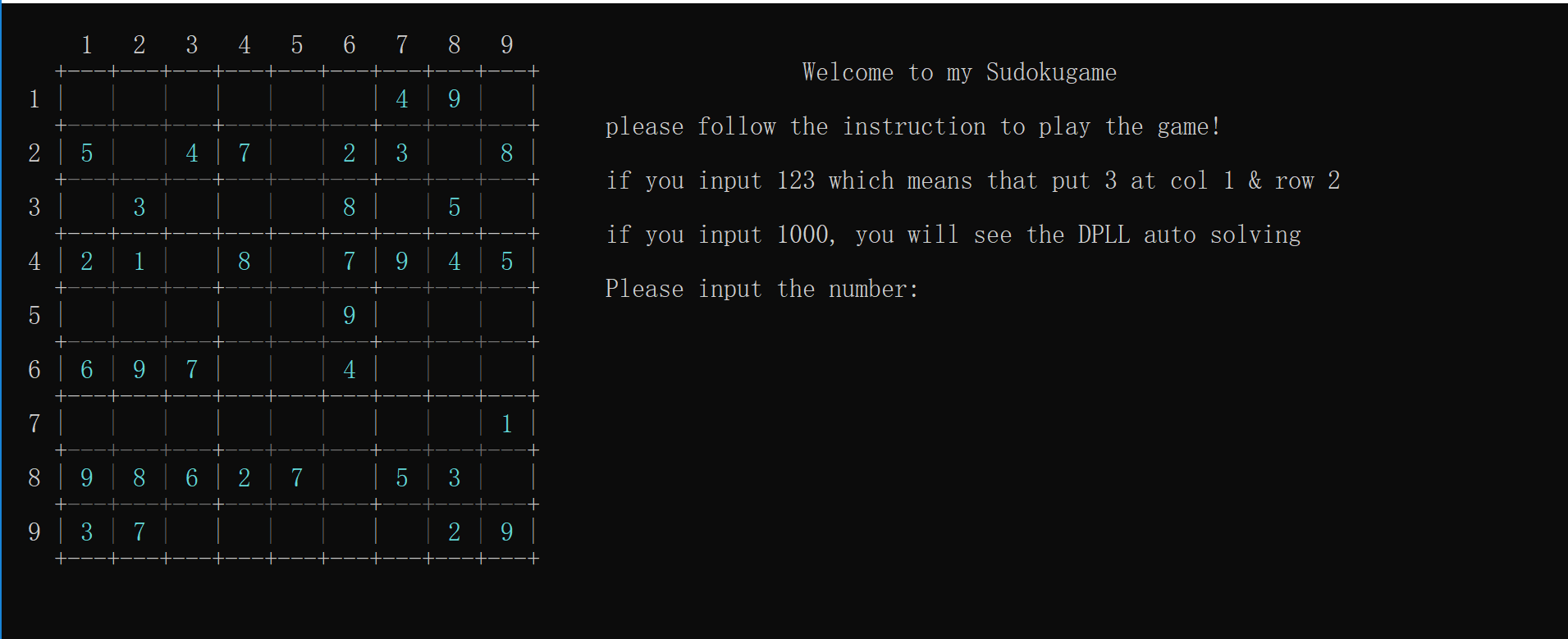


图1-3 输入2后界面示意图

* **以上为程序菜单的介绍**
  1. **求解cnf文件中的SAT问题介绍**
* **首先，在主菜单中输入1，上一节已经介绍**
* **然后在新界面输入你要处理的文件的绝对位置，如下图所示**

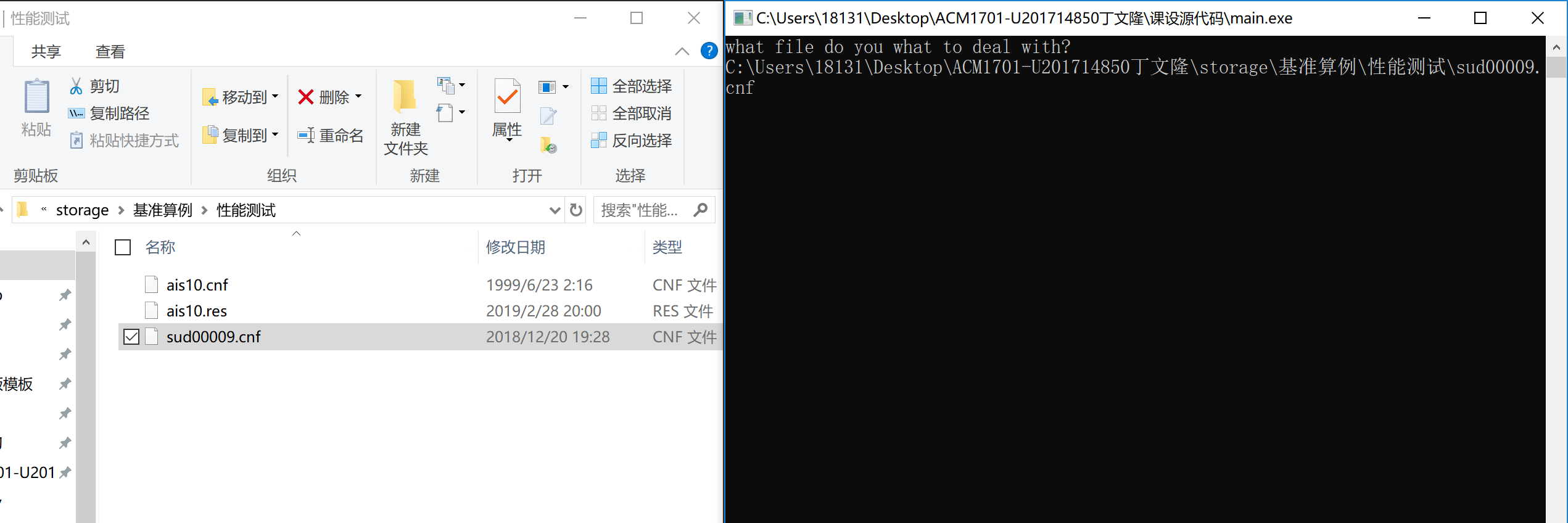


图2-1 输入处理文件的绝对位置示意图

* **点击回车，会发现屏幕上出现输出的链表，链表输出完成后，会显示这是否是一个满足的算例，如下图所示**

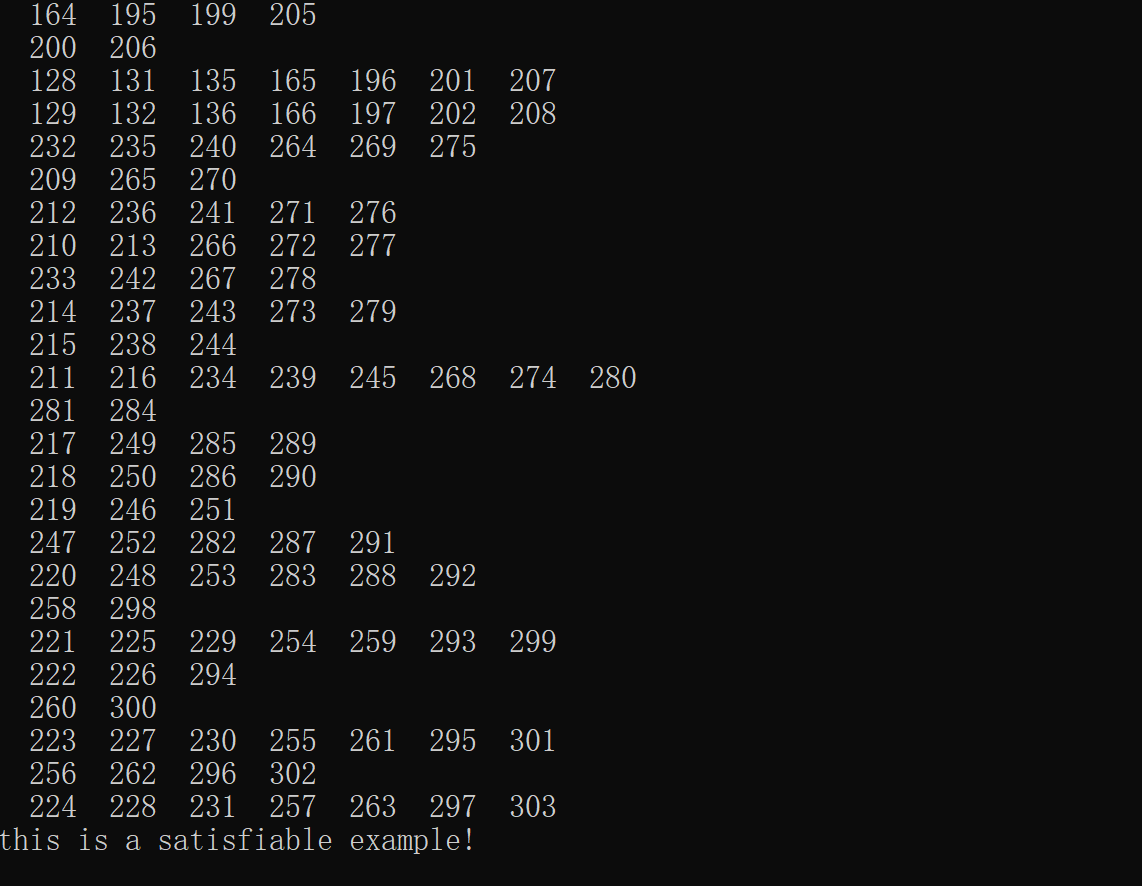


图2-2 输出链表以及结果示意图

由输出结果可知，这是一个满足算例

* **结束程序之后，你会发现res文件出现在了同一目录下，点开可查看输出信息，如下图所示：**

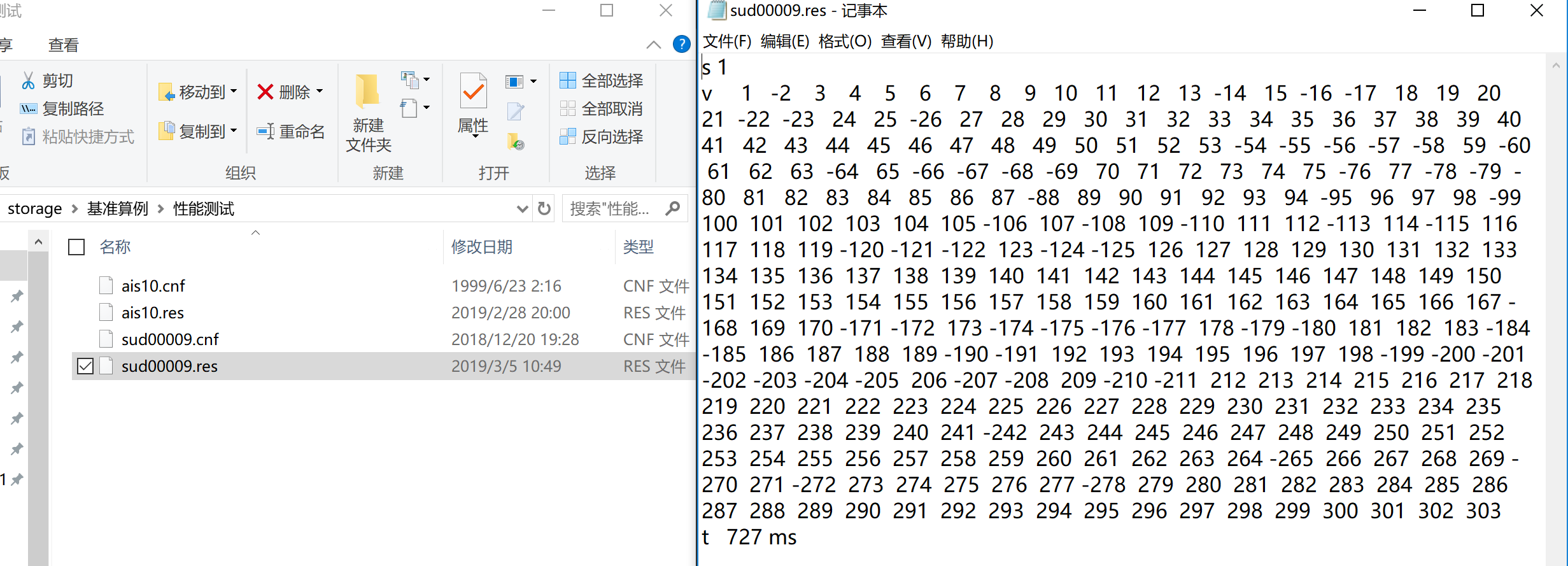


图2-3 对应res文件内容示意图

* 第一行的s代表是否是一个满足算例，若是则为1，不是则为0。
* v后面跟的一串数字代表每个文字取的布尔真值，是正取真，是负取假。
* t代表其中DPLL算法运行的时间。（注意并不是整个程序运行的时间，因为输出链表花费时间很长。）
* **以上是关于处理cnf文件中SAT问题的程序的操作介绍**

**三、数独游戏介绍**

* **在主菜单中输入2并回车，如第一节所示。**
* **你将会看到如下界面：**

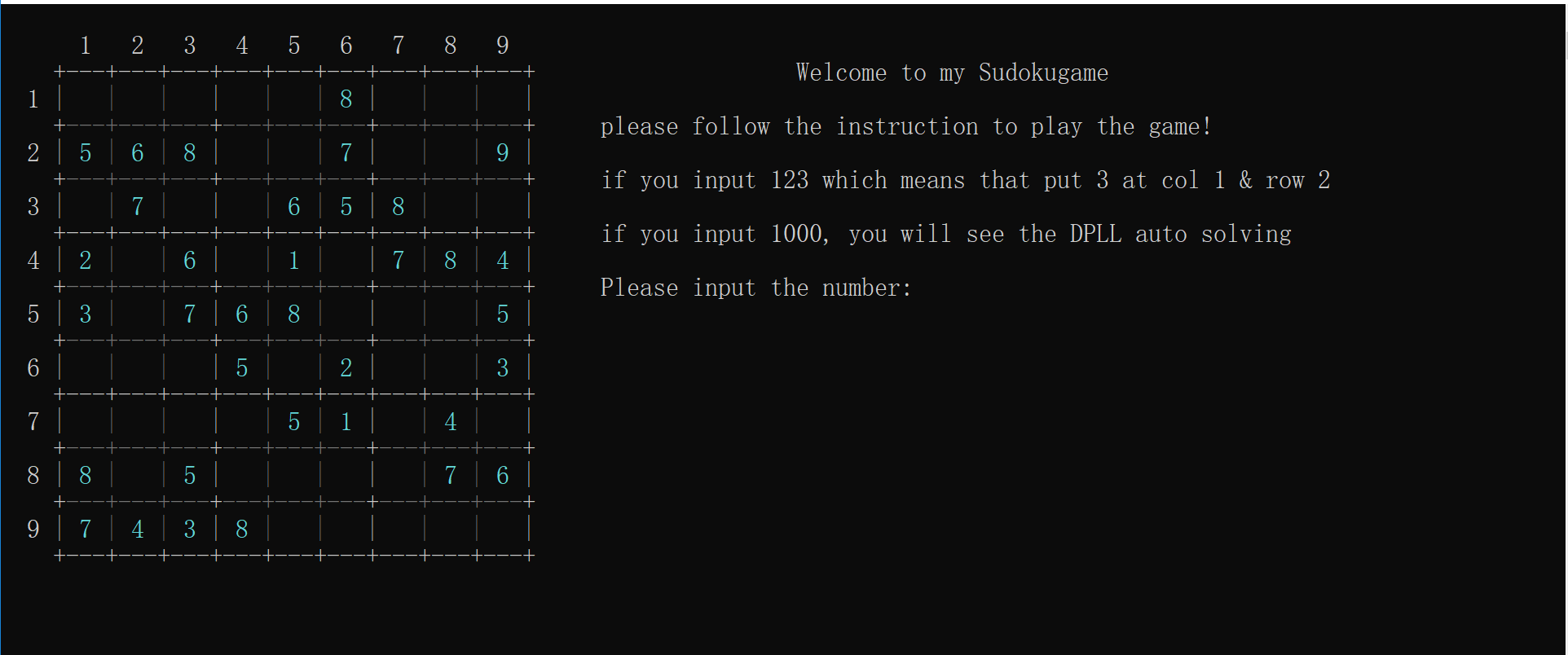


图3-1 数独游戏界面示意图

* **输出的cnf文件路径需要玩家自己设置，在main.cpp中改源代码，输入你想输出cnf文件的位置。**（PS：若玩家自己不更改则无法看到cnf文件）（PS：C语言语法要求，输入路径要双斜杠）下图仅为参考所用：

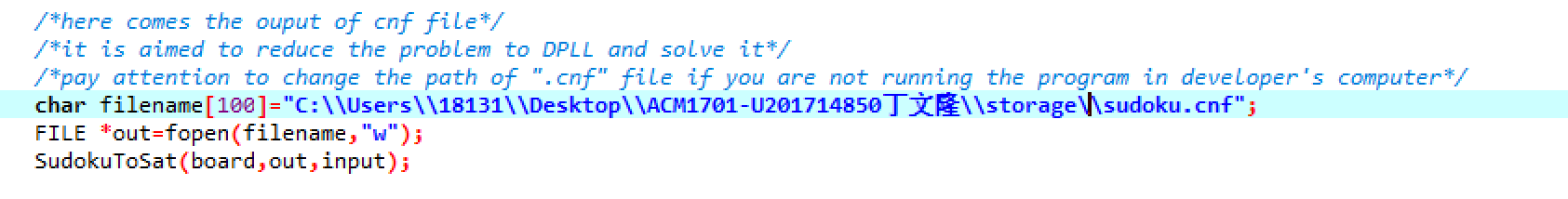


图3-2 修改cnf文件位置示意图

* **你可以看到输出的cnf文件如下：**

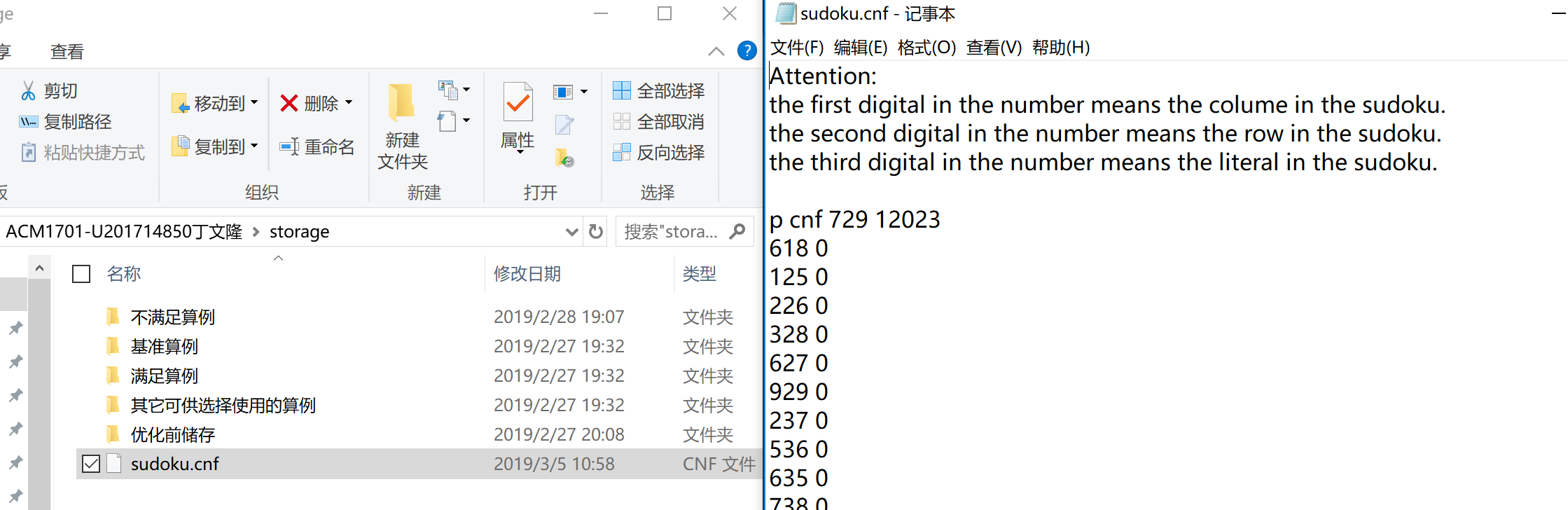


图3-3 输出的cnf文件示意图

* 回到数独游戏的玩法，正如提示所示，**输入一个三位数，如果想在第一行第2列输入9，则输入219（\*注意第一个数字是列，第二个数字是行）效果如下图所示：**

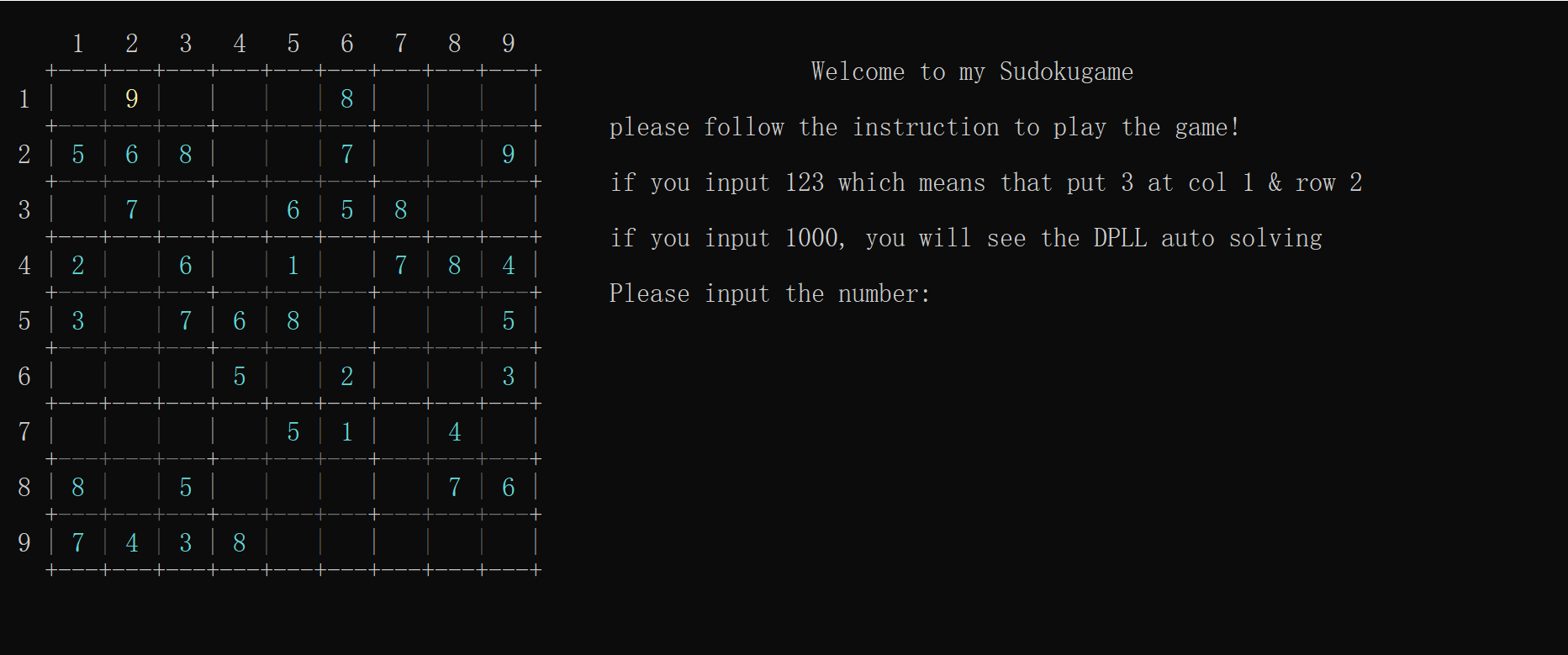


图3-4 用户输入正确结果效果图

你会发现黄色的9填入了第一行第二列，填完之后填数字的位子自动清空方便下一次填写。

* **如果你填入的数字不符合数独规则（即填入位置同行同列同块有相同数字），程序会自动提示你是错误的，如下图：**

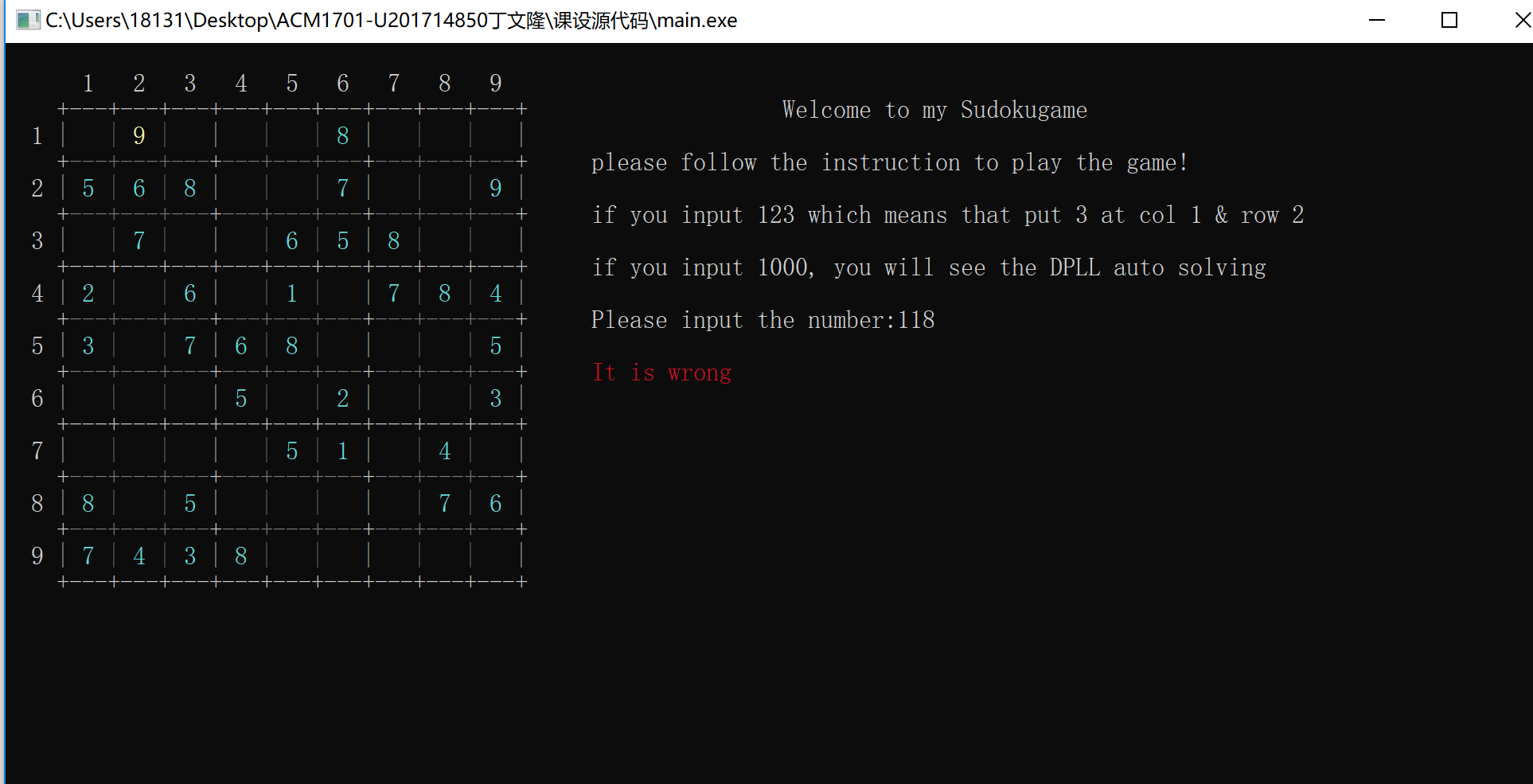


图3-5 用户输入不符合数独规则报错

作短暂停留之后又会清空方便下次填写。

* **如果你填入的位置已经有数字了，程序也会提醒你，如下图：**

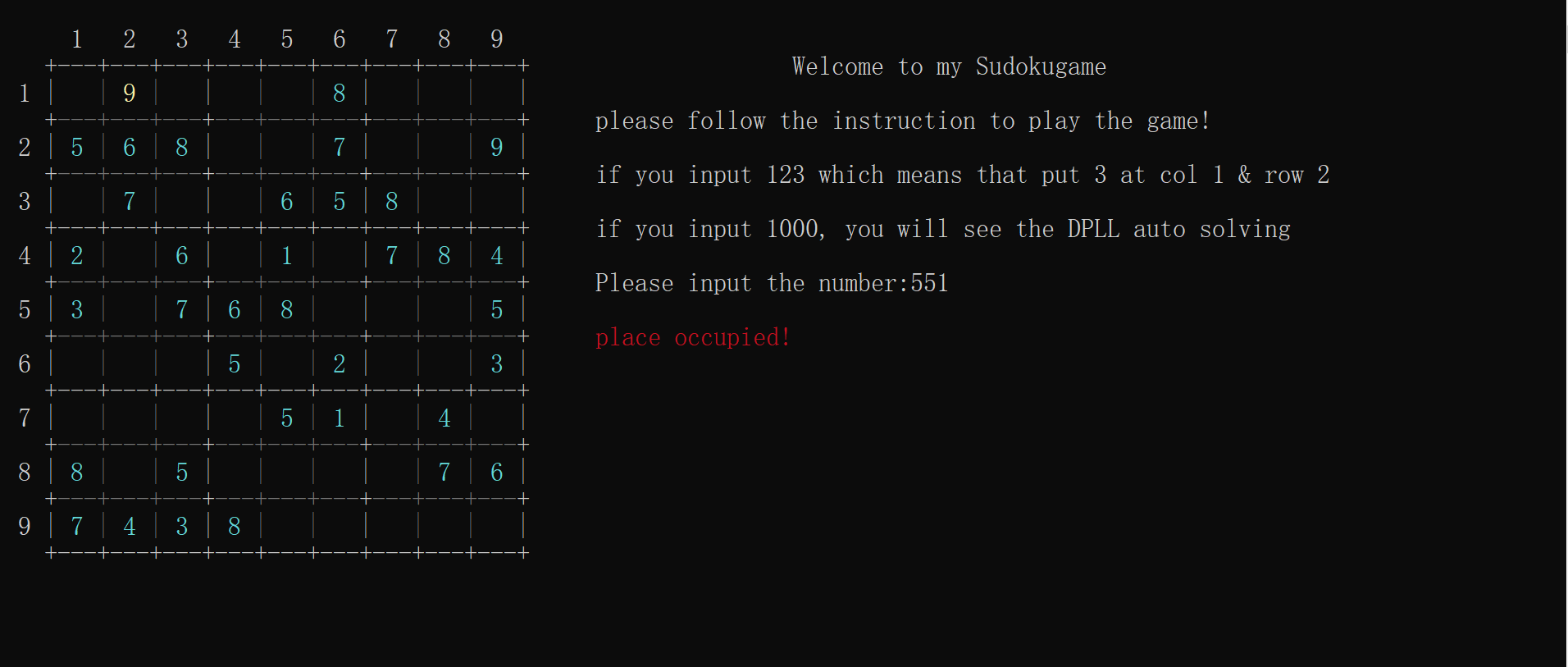


图3-6 用户输入位置被占用报错

作短暂停留之后又会清空方便下次填写。

* **如果想查看DPLL算法自动求解，只需要输入1000即可，程序会动态得显示答案（一个接一个地出现）求解完成截图如下：**

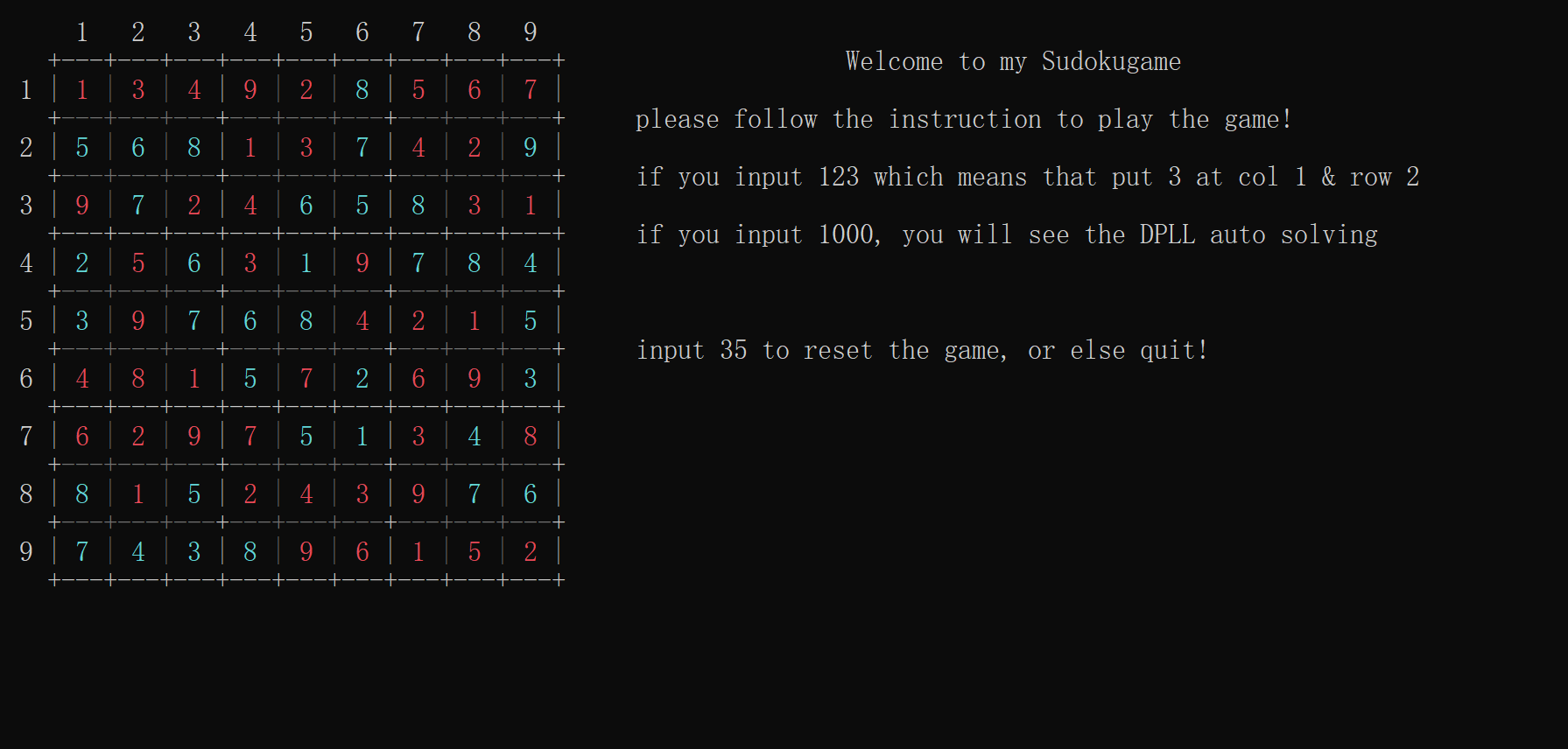


图3-7 DPLL自动求解效果示意图

* **如果想重新玩下一盘游戏，输入35（实际上输入35-55都是可行的，该输入设置的是初始填充的个数，只不过35个游戏体验最佳），若输入其他数字，则退出程序。如下图所示：**

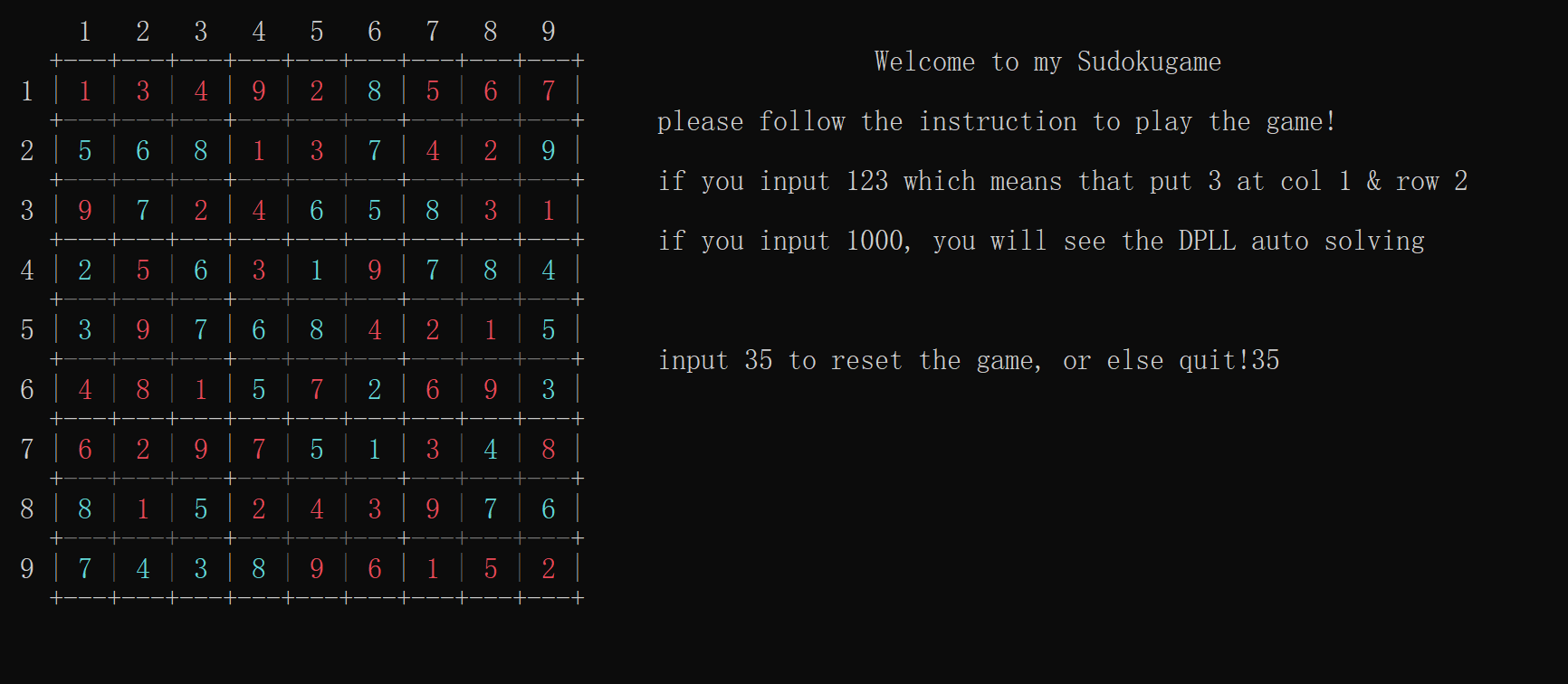


图3-8 输入重启游戏的数字

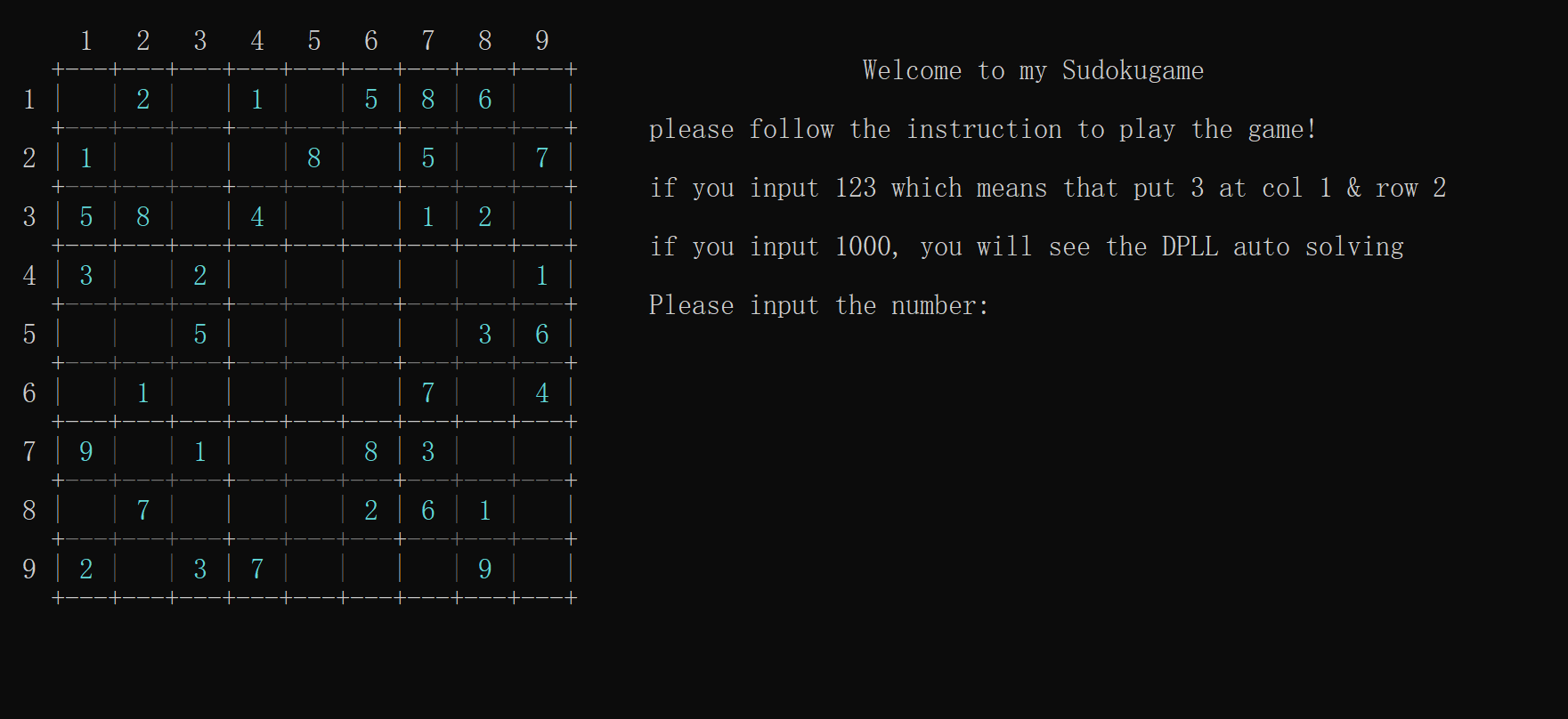


图3-9 重启后游戏界面

* **以上就是关于数独游戏的介绍。**