

# 实验报告

## 实验题目与要求

本次实验主要内容是实现一个简单的数独软件，具体要求如下

### 基本功能部分

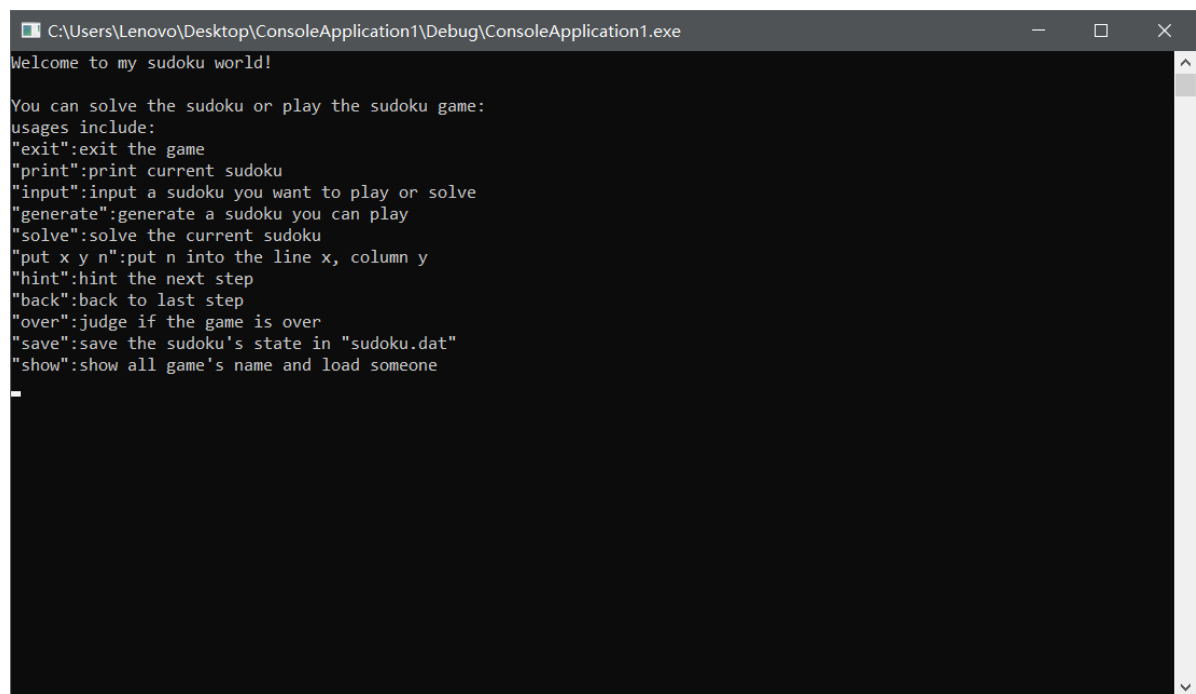
基本功能部分不需要通过图形化界面实现

下文提到的数独状态指9\*9的数独内容

下文使用的数独仅作为格式示例，并不作为内容实例

### 1.程序说明信息以及交互

程序运行后，输出提示信息，提示信息包括下面全部功能的输入说明



```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Welcome to my sudoku world!

You can solve the sudoku or play the sudoku game:
usages include:
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
Current Sudoku
0 1 4 0 0 0 0 0 8
0 0 0 0 0 6 0 0 0
0 0 0 0 4 0 9 0 0
0 0 0 0 0 0 2 0 6
0 0 0 0 1 0 5 0 0
0 0 7 0 0 8 0 0 0
0 0 0 0 0 0 0 0 0
7 0 0 3 0 0 0 0 9
0 0 0 0 0 1 3 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 2.选择游戏难度

游戏功能介绍后，将会让玩家选择游戏难度

F1 困难模式，F2正常模式，F3简单模式

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Now please choose the level of game if you want to play by pressing the keyboard:
"F1" Hard  "F2" Normal  "F3" Easy
Order:Level Choice:
HARD MODE
Press any key to continue . . .
```

## 3.打印数独

输入

print

输出9行当前的数独状态并换行，其中每行数据用空格作为分隔符。未填入位置用0表示

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
print
current sudoku state is
0 0 0 0 0 7 4 0 0
0 0 8 0 0 0 0 0 0
5 0 7 6 0 0 0 1 0
0 0 0 0 0 0 0 4 0
0 0 3 0 0 0 0 0 0
8 0 0 0 4 0 0 6 1
0 3 0 0 0 0 0 0 5
0 0 0 0 3 0 0 0 0
0 0 0 0 1 0 0 0 0
Press any key to continue . . .

Current Sudoku
0 0 0 0 0 7 4 0 0
0 0 8 0 0 0 0 0 0
5 0 7 6 0 0 0 1 0
0 0 0 0 0 0 0 4 0
0 0 3 0 0 0 0 0 0
8 0 0 0 4 0 0 6 1
0 3 0 0 0 0 0 0 5
0 0 0 0 3 0 0 0 0
0 0 0 0 1 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 4.输入数独

输入

input

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
input
Please input a sudoku:
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 3 4 6 1 7 0 0
0 0 8 0 0 0 4 0 0
0 0 6 0 5 0 1 0 0
0 0 4 0 0 0 8 0 0
0 0 1 6 7 9 2 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0

Current Sudoku
0 0 0 0 0 7 4 0 0
0 0 8 0 0 0 0 0 0
5 0 7 6 0 0 0 1 0
0 0 0 0 0 0 0 4 0
0 0 3 0 0 0 0 0 0
8 0 0 0 4 0 0 6 1
0 3 0 0 0 0 0 0 5
0 0 0 0 3 0 0 0 0
0 0 0 0 1 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

如果输入是合法的（输入全是数字字符，且已填入位置的行列对角线九宫格不重复），存储输入到数独状态，输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
input
Please input a sudoku:
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 3 4 6 1 7 0
0 0 8 0 0 0 4 0
0 0 6 0 5 0 1 0
0 0 4 0 0 0 8 0
0 0 1 6 7 9 2 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
accept your input
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 3 4 6 1 7 0
0 0 8 0 0 0 4 0
0 0 6 0 5 0 1 0
0 0 4 0 0 0 8 0
0 0 1 6 7 9 2 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
Press any key to continue . . .

Current Sudoku
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 3 4 6 1 7 0
0 0 8 0 0 0 4 0
0 0 6 0 5 0 1 0
0 0 4 0 0 0 8 0
0 0 1 6 7 9 2 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

否则数独状态不应发生改变，并且输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
input
Please input a sudoku:
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 3 4 6 1 7 0
0 0 8 0 0 0 4 0
0 0 6 0 5 0 1 0
0 0 4 0 0 0 8 0
0 0 1 6 7 9 2 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 2
illegal input
Press any key to continue . . .

Current Sudoku
0 1 0 0 0 9 0 0
0 0 0 0 0 0 0 0
6 0 0 0 0 0 9 0
0 0 0 0 0 0 0 0
0 0 0 0 0 2 0 0
0 0 0 0 0 0 9 0
0 3 5 7 0 4 0 0
7 2 1 0 6 5 0 0
4 0 6 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

输入的数独需要至少17个非零数字，可以有多解

## 5.生成数独

输入

generate

程序设置数独状态为你生成的数独，并打印数独

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
generate
generated sudoku is
0 0 0 0 0 7 0 0
4 0 0 0 0 0 0 0
0 0 0 0 7 0 0 5
0 0 4 0 0 3 0 0
0 3 0 0 0 0 0 4
0 8 0 0 0 0 0 3
0 0 0 6 0 0 3 0
0 0 0 0 4 1 9 0
0 0 0 3 0 0 4 0
Current Sudoku
0 1 0 0 0 9 0 0
0 0 0 0 0 0 0 0
6 0 0 0 0 0 9 0
0 0 0 0 0 0 0 0
0 0 0 0 0 2 0 0
0 0 0 0 0 0 9 0
0 3 5 7 0 4 0 0
7 2 1 0 6 5 0 0
4 0 6 0 0 0 0 0
Press any key to continue . . .

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

注意，生成的数独需要至少17个非零数字，可以有多解或没有解，但必须是一个合法的数独（已填入位置的行列对角线九宫格不重复）

注意，这里合法的数独也可以是没有解的。

## 6.输出数独的解

输入

solve

如果数独有解，输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
solve
sudoku before solution is
0 0 0 0 0 7 0 0
4 0 0 0 0 0 0 0
0 0 0 0 7 0 0 5
0 0 4 0 0 3 0 0
0 3 0 0 0 0 0 4
0 8 0 0 0 0 0 3
0 0 0 6 0 0 3 0
0 0 0 0 4 1 9 0
0 0 0 3 0 0 4 0
Current Sudoku
0 0 0 0 0 7 0 0
4 0 0 0 0 0 0 0
0 0 0 0 7 0 0 5
0 0 4 0 0 3 0 0
0 3 0 0 0 0 0 4
0 8 0 0 0 0 0 3
0 0 0 6 0 0 3 0
0 0 0 0 4 1 9 0
0 0 0 3 0 0 4 0
one possible solution is
1 2 3 4 5 6 7 8 9
4 7 5 8 1 9 6 2 3
8 6 9 3 2 7 4 1 5
7 1 4 5 6 3 2 9 8
5 3 6 9 8 2 1 7 4
9 8 2 1 7 4 5 3 6
2 4 7 6 9 8 3 5 1
3 5 8 2 4 1 9 6 7
6 9 1 7 3 5 8 4 2
Press any key to continue . . .

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

不需要输出所有的解。

如果数独无解，输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
solve
sudoku cannot be solved
Press any key to continue . . .
Current Sudoku
0 5 0 0 3 0 0 0 0
0 0 1 6 0 0 9 0 8
3 4 0 7 0 1 0 0 0
0 0 0 0 0 7 0 0 5
0 9 0 0 0 0 0 8 0
7 0 0 2 0 0 0 0 0
0 0 0 3 0 9 0 6 1
6 0 4 0 0 8 3 0 0
0 0 0 0 6 0 0 7 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

注意，在解数独和hint功能的测试中，我们使用的测试样例一定是有解的数独。

## 7.填入数字合法情况

采用下列输入格式向数独中填入数字

向第1行第3列填入2，

```
put 1 3 2
```

注意，输入的行列下标从1开始到9结束，都是合法的

对于大部分正常情况，输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 3 2
Last Sudoku
now sudoku is.
3 1 2 0 0 9 0 6 0
0 0 0 0 0 4 0 2
0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 6
0 0 9 0 1 0 0 0
0 0 7 0 0 0 0 0
0 0 5 0 0 0 2 0
0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0

3 1 0 0 0 9 0 6 0
0 0 0 0 0 4 0 2
0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 6
0 0 9 0 1 0 0 0
0 0 7 0 0 0 0 0
0 0 5 0 0 0 2 0
0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 填入数字异常情况

### 1. 如果试图修改题目固定的数字

```
put 1 1 1
```

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 1 1
ERROR there are already a number in this position
Press any key to continue . . .

Current Sudoku
3 1 0 0 0 9 0 6 0
0 0 0 0 0 0 4 0 2
0 0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 0 6
0 0 9 0 1 0 0 0 0
0 0 7 0 0 0 0 0 0
0 0 5 0 0 0 0 2 0
0 0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

### 2. 如果填入的数字过大

```
put 1 3 99
```

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 3 99
ERROR data range exceed
Press any key to continue . . .

Current Sudoku
3 1 0 0 0 9 0 6 0
0 0 0 0 0 0 4 0 2
0 0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 0 6
0 0 9 0 1 0 0 0 0
0 0 7 0 0 0 0 0 0
0 0 5 0 0 0 0 2 0
0 0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

### 3. 如果填入的数字过小

```
put 1 3 -2
```

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 3 -2
ERROR the number is under data range
Press any key to continue . . .

Current Sudoku
3 1 0 0 9 0 6 0
0 0 0 0 0 4 0 2
0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 6
0 0 9 0 1 0 0 0
0 0 7 0 0 0 0 0
0 0 5 0 0 0 2 0
0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

#### 4. 如果输入的填入位置不合法

```
put 1 99 1
```

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 99 1
ERROR illegal location
Press any key to continue . . .

Current Sudoku
3 1 0 0 9 0 6 0
0 0 0 0 0 4 0 2
0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 6
0 0 9 0 1 0 0 0
0 0 7 0 0 0 0 0
0 0 5 0 0 0 2 0
0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

#### 5. 如果输入的填入位置不合法输出具体的重复信息，说明重复数字所在位置

(1) 不满足行列数字不同

```
put 1 3 6
```



```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 1 3 6
ERROR line illegal
The line 1 column 8 has the same number
Press any key to continue . . .

Current Sudoku
3 1 0 0 0 9 0 6 0
0 0 0 0 0 0 4 0 2
0 0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 0 6
0 0 9 0 1 0 0 0 0
0 0 7 0 0 0 0 0 0
0 0 5 0 0 0 0 2 0
0 0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

(2) 不满足对角线元素不同

put 9 9 3

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 9 9 3
ERROR cross illegal
The line 1 column 1 has the same number
Press any key to continue . . .

Current Sudoku
3 1 0 0 0 9 0 6 0
0 0 0 0 0 0 4 0 2
0 0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 0 6
0 0 9 0 1 0 0 0 0
0 0 7 0 0 0 0 0 0
0 0 5 0 0 0 0 2 0
0 0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

(3) 不满足3x3小格内元素各不相同

put 3 5 9

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\SUDOKU.exe
Order:
put 3 5 9
ERROR 3x3 illegal
3x3 sub table has the same number
Press any key to continue . . .

Current Sudoku
3 1 0 0 9 0 6 0
0 0 0 0 0 4 0 2
0 0 0 0 3 0 0 0
0 0 3 5 0 0 0 6
0 0 9 0 1 0 0 0
0 0 7 0 0 0 0 0
0 0 5 0 0 0 2 0
0 0 0 0 0 8 0 0
0 0 0 9 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 9.数独提示信息

输入

hint

### 1. 如果有解

程序选择一个未填入数字的空格，并填入一个非零数字作为提示信息，保存到数独状态后打印输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
hint
Last Sudoku
before hint, sudoku is
3 1 4 2 0 9 7 0 0
5 9 8 1 0 6 0 3 0
0 7 2 0 0 3 9 0 0
1 4 3 0 9 7 2 0 6
0 6 0 0 0 2 5 0 0
0 5 7 6 0 8 1 9 4
9 0 0 0 8 0 6 2 0
7 2 0 3 6 5 8 4 0
4 8 6 9 0 0 3 0 7
now it is
3 1 4 2 5 9 7 0 0
5 9 8 1 0 6 0 3 0
0 7 2 0 0 3 9 0 0
1 4 3 0 9 7 2 0 6
0 6 0 0 0 2 5 0 0
0 5 7 6 0 8 1 9 4
9 0 0 0 8 0 6 2 0
7 2 0 3 6 5 8 4 0
4 8 6 9 0 0 3 0 7
Press any key to continue . . .

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

注意，对于所有hint之前有解的数独，hint之后的数独也有解。（也就是说不能乱提示）。可以通过不停的hint获得最终的解。

### 2.如果没解

提示数独无解

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
hint
Last Sudoku
before hint, sudoku is
0 5 0 0 3 0 0 0 0
0 0 1 6 0 0 9 0 8
3 4 0 7 0 1 0 0 0
0 0 0 0 0 7 0 0 5
0 9 0 0 0 0 0 8 0
7 0 0 2 0 0 0 0 0
0 0 0 3 0 9 0 6 1
6 0 4 0 0 8 3 0 0
0 0 0 6 0 0 7 0
sudoku cannot be solved.
Press any key to continue . . .
Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

### 3.如果数独填满了

提示游戏结束了

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
hint
Last Sudoku
before hint, sudoku is
1 2 3 4 5 6 7 9 8
4 7 5 8 1 9 2 6 3
8 9 6 3 2 7 4 1 5
2 1 4 5 6 3 8 7 9
5 3 7 1 9 8 6 2 4
6 8 9 2 7 4 5 3 1
9 4 1 6 8 2 3 5 7
3 5 2 7 4 1 9 8 6
7 6 8 9 3 5 1 4 2
Game is over!
You can put "over" to end the game
Press any key to continue . . .
Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 10.回到上一步

由于玩家在游戏过程中，可能认为自己的上一步填法有问题，可以使用back指令，回到上一步

注意，当返回到游戏开始状态时，就不可以再返回了

输入

back

### 1. 成功返回

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
back
the last sudoku is:
1 3 7 4 6 8 0 5 0
5 0 9 0 0 2 7 4 0
2 4 6 0 0 5 0 0 0
8 2 0 3 0 0 0 9 0
3 6 5 9 2 7 0 8 0
0 7 0 0 8 4 6 3 2
6 5 3 0 7 1 9 2 4
4 0 2 6 0 9 3 7 8
0 9 0 2 4 3 0 0 5
Press any key to continue . . .

Current Sudoku
1 3 7 4 6 8 2 5 0
5 0 9 0 0 2 7 4 0
2 4 6 0 0 5 0 0 0
8 2 0 3 0 0 0 9 0
3 6 5 9 2 7 0 8 0
0 7 0 0 8 4 6 3 2
6 5 3 0 7 1 9 2 4
4 0 2 6 0 9 3 7 8
0 9 0 2 4 3 0 0 5

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 2. 已到初始状态

将不改变数独，输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
back
you cannot back.
Press any key to continue . . .

Current Sudoku
0 0 2 0 0 0 0 0 0
8 0 0 0 0 0 0 0 0
0 0 9 2 0 0 0 0 0
0 2 8 6 7 0 0 0 4
9 0 4 0 0 0 0 0 0
0 0 0 0 0 4 1 0 0
2 0 0 0 0 0 0 1 5
0 3 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 11.保存游戏状态

由于有些数独很难，可能一时半会儿解不出来，可以使用save 命令保存当前的数独状态

输入

```
save
```

提示输入记录的名称

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
save
input your game record name

Current Sudoku
1 2 0 0 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 0 0 9 0 5 1 4 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 1. 输入自定义的记录名称

```
test
```

输出,保存成功

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
save
input your game record name
test
your game state has been recorded successfully
Press any key to continue . . .

Current Sudoku
1 2 0 0 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 0 0 9 0 5 1 4 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

将当前的数独状态，记录名称，当前时间保存到同一目录的文件下

## 2. 如果输入的记录名称重复

应该输出提示信息，程序回到接收记录名称输入状态

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
save
input your game record name
test
This name has existed
input your game record name again
-

Current Sudoku
1 0 3 4 5 0 0 9 8
4 7 5 8 0 9 2 6 3
8 0 0 0 2 0 4 0 0
0 1 0 5 0 3 8 7 0
0 3 7 1 9 8 6 0 4
0 0 0 2 7 0 5 3 1
9 0 0 0 8 0 3 5 7
3 0 0 0 4 1 0 8 6
7 6 8 9 0 5 0 0 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

### 3. 重新输入

test2

保存成功

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
save
input your game record name
test
This name has existed
input your game record name again
test1
your game state has been recorded successfully
Press any key to continue . . .

Current Sudoku
1 0 3 4 5 0 0 9 8
4 7 5 8 0 9 2 6 3
8 0 0 0 2 0 4 0 0
0 1 0 5 0 3 8 7 0
0 3 7 1 9 8 6 0 4
0 0 0 2 7 0 5 3 1
9 0 0 0 8 0 3 5 7
3 0 0 0 4 1 0 8 6
7 6 8 9 0 5 0 0 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 12. 加载游戏状态

可以查看已保存的游戏状态，输入show指令读档

### 1. 显示存档

输入

show

输出

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
show
0-new game
1-test-2021.6.26.16:33
2-test1-2021.6.26.16:35

Current Sudoku
1 0 3 4 5 0 0 9 8
4 7 5 8 0 9 2 6 3
8 0 0 0 2 0 4 0 0
0 1 0 5 0 3 8 7 0
0 3 7 1 9 8 6 0 4
0 0 0 2 7 0 5 3 1
9 0 0 0 8 0 3 5 7
3 0 0 0 4 1 0 8 6
7 6 8 9 0 5 0 0 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

## 2. 读档

输入1后恢复游戏进度，打印所保存的数独状态

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
show
0-new game
1-test-2021.6.26.16:33
2-test1-2021.6.26.16:35
1
last time sudoku is
1 2 0 0 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 0 0 9 0 5 1 4 2
Press any key to continue . . .

Current Sudoku
1 0 3 4 5 0 0 9 8
4 7 5 8 0 9 2 6 3
8 0 0 0 2 0 4 0 0
0 1 0 5 0 3 8 7 0
0 3 7 1 9 8 6 0 4
0 0 0 2 7 0 5 3 1
9 0 0 0 8 0 3 5 7
3 0 0 0 4 1 0 8 6
7 6 8 9 0 5 0 0 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
print
current sudoku state is
1 2 0 0 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 0 0 9 0 5 1 4 2
Press any key to continue . . .

Current Sudoku
1 2 0 0 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 0 0 9 0 5 1 4 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

### 13.判断游戏是否结束

#### 1. 玩家成功解完数独

##### 游戏胜利

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
over
Congratulations!
You win!
Please input "exit" to end the game
Press any key to continue . . .

Current Sudoku
2 1 3 4 7 5 6 8 9
7 5 6 8 3 9 1 2 4
4 8 9 2 6 1 3 5 7
1 6 8 3 9 4 2 7 5
9 3 7 5 1 2 4 6 8
5 2 4 6 8 7 9 3 1
3 9 5 7 4 6 8 1 2
6 7 1 9 2 8 5 4 3
8 4 2 1 5 3 7 9 6

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

#### 2. 如果数独本身无解，玩家自行判断游戏结束

##### 游戏胜利



```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
over
Congratulations!
You win and this sudoku is not solvable!
Please input "exit" to end the game
Press any key to continue . . .

Current Sudoku
0 5 0 0 3 0 0 0 0
0 0 1 6 0 0 9 0 8
3 4 0 7 0 1 0 0 0
0 0 0 0 0 7 0 0 5
0 9 0 0 0 0 0 8 0
7 0 0 2 0 0 0 0 0
0 0 0 3 0 9 0 6 1
6 0 4 0 0 8 3 0 0
0 0 0 0 6 0 0 7 0

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

3. 如果数独本身有解，由于玩家错误填写导致数独无解

### 游戏没有结束

提示如下

```
C:\Users\Lenovo\Desktop\ConsoleApplication1\Debug\ConsoleApplication1.exe
Order:
over
Game is not over ,maybe you have made some mistakes
Press any key to continue . . .

Current Sudoku
1 2 3 4 5 0 7 9 8
4 7 5 0 1 9 2 6 3
8 0 0 3 2 7 4 1 0
2 0 4 5 6 0 0 7 9
5 0 7 0 0 8 6 0 0
0 8 0 2 0 0 5 0 0
9 0 1 0 8 0 3 5 0
3 5 0 7 4 1 0 8 6
0 3 0 9 0 5 1 4 2

Usage
"exit":exit the game
"print":print current sudoku
"input":input a sudoku you want to play or solve
"generate":generate a sudoku you can play
"solve":solve the current sudoku
"put x y n":put n into the line x, column y
"hint":hint the next step
"back":back to last step
"over":judge if the game is over
"save":save the sudoku's state in "sudoku.dat"
"show":show all game's name and load someone
```

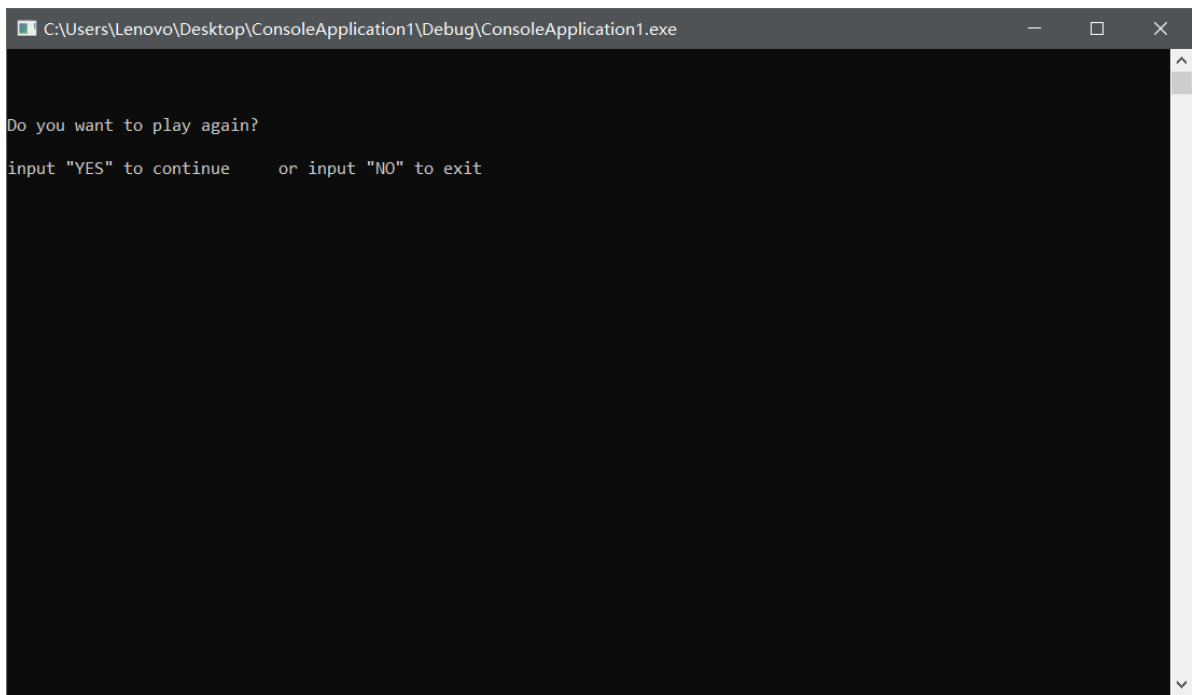
## 14.选择游戏是否重新开始

玩家选择退出游戏，询问玩家是否重新开始游戏

输入

exit

输出

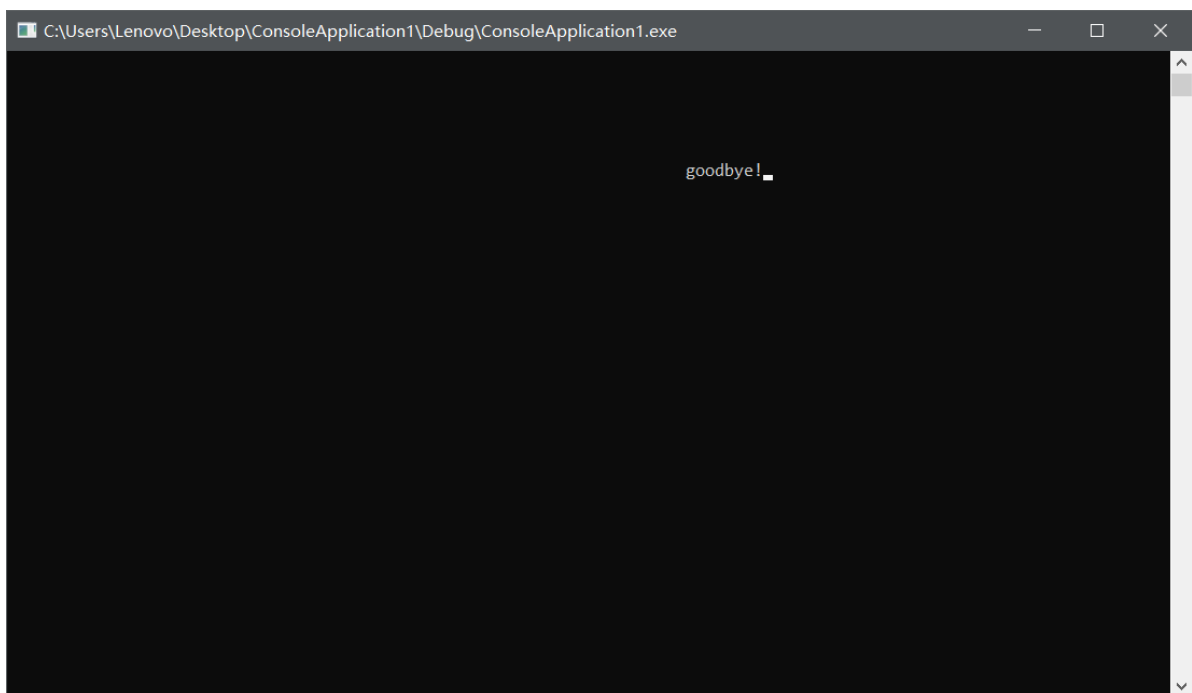


1.选择yes

重新开始游戏

2.选择no

输出



并等待5秒后关闭命令行窗口

## 具体设计

---

## 整体思路

在头文件中创建好关于数独的结构体 Sudoku，方便通过**指针传递**进行整体的处理

```
typedef struct SudokuStruct {  
    int currTab[9][9];  
    int initTab[9][9];  
    bool isInitSolvable;  
    bool isCurrSolvable;  
} Sudoku;  
//Sudoku *S 在main函数中建立
```

通过stringProceed函数与orderReceieve函数，接收并处理来自用户的指令

包括：print, generate, input, solve, hint, put, save, show, back, over

然后根据不同的指令对应各个功能的具体实现，并对输入各个情况进行反馈，达到**交互式的功能**。

```
int stringProceed(Sudoku *S);
```

stringProceed处理来自命令行的各个指令，并在处理后直接指向orderReceieve函数

```
void orderReceive(Sudoku *S, int order, int *number)
```

orderReceive接收各种指令，每种指令（int order）对应一个数字，number指针指向含3个int值的数组，其中储存可能来自用户指令“put”中数独的坐标和修改的值。

orderReceive可以通过这些指令，指向各个具体函数，以达到功能的实现

orderReceive可以更具具体函数的返回值，输出相应的提示信息

例如如下，stringProceed当接收到input的指令时，传递给orderReceive，其中对应的函数是inputTab(S)，当函数返回1时，输出"accept your input"，并打印数独；当函数返回0时，提示是非法输入

```
//int stringProceed(Sudoku *S)  
if (!strcmp(string, "input")) {  
    orderReceive(S, INPUT_TAB, number);  
    return INPUT_TAB;  
}  
//void orderReceive(Sudoku *S, int order, int *number)  
case INPUT_TAB:  
    printf("Please input a sudoku:\n");  
    if (inputTab(S)) {  
        printf("accept your input\n");  
        print(S->initTab);  
    } else {  
        printf("illegal input\n");  
    }  
    break;
```

# 具体功能的实现

## 1.程序说明信息

```
void pos(int x, int y)
```

该函数可以改变光标位置，以达到任意位置输出的目的

```
void welcome();
```

为用户提供数独各项功能的指令和具体作用等

如下，告诉用户数独的基本功能

```
pos(0, 2);
printf("You can solve the sudoku or play the sudoku game:\n");
for (i = 0; i <= 2500; i += 100) {
    sleep(100);
    if (GetAsyncKeyState(VK_RETURN))
        break;
}
```

## 2.游戏难度的选择

```
void welcome()
```

在该函数中包含了游戏难度选择

```
while (1) {
    if (GetAsyncKeyState(VK_F1)) {
        printf("HARD MODE\n");
        level = HARD;
        break;
    } else if (GetAsyncKeyState(VK_F2)) {
        printf("NORMAL MODE\n");
        level = NORMAL;
        break;
    } else if (GetAsyncKeyState(VK_F3)) {
        printf("EASY MODE\n");
        level = EASY;
        break;
    }
}
```

通过用户的键盘输入，改变level的值,在游戏时，就可以改变数独的初始个数以达到改变难度的目的

## 3.打印数独

```
void print(int table[][9]);
```

输入9\*9数组，打印数独

## 4.输入数独

```
int inputTab(Sudoku *S);
```

读取输入的数独，初始化当前数独的状态

如果数独是合法输入，初始化Sudoku \*S对应的结构体中的值，并返回1

如果数独是不合法的，不改变数独，返回0

## 5.生成数独

根据一开始选择的难度随机生成一个可解的数独，初始化数独的状态

```
void randomInit(Sudoku *S)
```

具体是通过随机选择数独库中的某一个数独，再随机消去一些位置上的数，达到生成数独的目的

## 6.输出数独的解 < 核心代码 >

设计思路：递归法 解数独的值

具体实现如下：

```
bool onesolution(Sudoku *S, int pos) {
    bool key = false;
    bool flag;
    int i = pos / 9;
    int j = pos % 9;
    if (S->currTab[i][j] != 0) { //已有原始数据
        if (pos == 80)
            key = true;
        else
            key = onesolution(S, pos + 1);
    } else {
        for (int k = 1; k <= 9; ++k) {
            S->currTab[i][j] = k;
            flag = isLegal(S);
            if (flag) {
                if (pos == 80)
                    key = true;
                else
                    key = onesolution(S, pos + 1);
            }
            if (key)
                break;
            else
                S->currTab[i][j] = 0;
        }
    }
    return key;
}
```

输入当前数独的状态和当前出发的位置 (pos/9代表行位置, pos%9代表列位置)

返回true 代表从此位置出发，当前数独存在一个可解值，可以探索到最后的位置

返回false 代表从当前数独的状态和位置出发，不存在一条可解路径  
在该函数的作用下，数独的当前状态会发生改变。

## 7.填入数字合法情况

```
int inputNum(Sudoku *S, int row, int col, int number)
```

输入待改变的行，列，和数字

返回1 代表合法输入，更新数独的状态

## 8.填入数字异常情况

```
int inputNum(Sudoku *S, int row, int col, int number)
```

输入待改变的行，列，和数字

返回0 代表填入的数字和已有数字重复（不满足同行、同列、对角线、3x3格子元素各不相同）

输出与其冲突的具体坐标

返回2 代表填入位置不合法

返回3 代表填入的数字过大

返回4 代表填入的数字过小

返回5 代表试图修改题目固定的数字

具体的输出语句在orderReceive中实现交互。

```
void orderReceive(Sudoku *S, int order, int *number)
{
    case INPUT_NUMBER:
        int inputNumberState;
        inputNumberState = inputNum(S, number[0], number[1], number[2]);
}
```

## 9.数独提示信息

```
int hint(Sudoku *S)
```

如果当前数独状态已经完全填满

说明玩家已经解完数独，返回 2；

如果当前数独状态没有填满，而数独是可解的

修改数独的状态，提示下一步，返回 1；

如果当前数独是无解的

返回 0；

hint函数中包含oneStepForward函数

```
int oneStepForward(Sudoku* S)
```

如果当前数独可解，修改一格数独当前的状态，返回 1；

如果当前数独不可解，不改变数独，返回 2；

## 10.回到上一步

设计思路：利用标准库栈函数

```
int back(Sudoku* S)
```

如果栈为空，说明回到最初状态，返回 0；

如果栈不为空，取出栈的top () 值，修改数独的状态，返回 1；

具体实现

```
int back(Sudoku* S) {
    int* position;
    if (st.empty())
        return 0;
    else {
        position = st.top();
        S->currTab[position[0] - 1][position[1] - 1] = position[2];
        free(position);
        st.pop();
        return 1;
    }
}
```

position指向一个含三个int的数组,position[0]代表行，position[1]代表列，position[2]代表具体数值；

```
cstack<Pos> st;
```

st是全局变量

st.push(position )在填入数字，提示数字时使用

## 11.保存游戏状态

设计思路：使用文件+链表

```
int save(Sudoku* S)
```

如果文件“sudoku.dat”不存在，创建一个二进制格式的文件，再存档；

如果文件存在，直接存档。

询问用户记录本次游戏名称，并提示用户输入

存档成功，记录下名称，数独的状态，时间，并返回 1

如果记录名称重复，提示信息重新输入

## 12.加载游戏状态

```
int show(Sudoku* s)
```

查看已保存的游戏状态

输出示例

```
0-new game
1-test-2021.6.26.16:33
2-test1-2021.6.26.16:35
```

输入0，重新开始新游戏

输入1，读档，改变数独的状态

异常输入

输入3，提示信息，重新输入

## 13.判断游戏是否结束

```
int isOver(Sudoku* s)
```

判断游戏是否结束，并更新S->isCurrSolvable的值

如果游戏没有结束，返回 0；

游戏结束情况：

如果初始数独是不可解的，玩家成功判断出数独无解，返回 1

如果初始数独是可解的，同时玩家填满数独，返回2

## 14.选择游戏是否重新开始

```
char Decide()
```

询问用户是否重新开始游戏

用户选择yes/YES 返回 y，主函数中循环；

选择no/NO 返回n，退出游戏，

输出：

```
Goodbye!
```

并在5秒后关闭游戏窗口



## 15.其他函数的说明

```
bool isSolvable(const Sudoku* s)
```

检查数独是否有解，不改变数独的状态。有解返回true，无解返回false

```
bool isLegal(Sudoku* s)
```

检查数独是否合法，满足同行、同列、对角线、3x3格子元素各不相同。合法返回true，非法返回false

```
bool isFull(Sudoku* s)
```

检查数独是否填满。填满返回true，没有填满返回false

```
sudokuFile* creatAccount(sudokuFile* head, char* fileName)
```

如果文件不存在，创建一个账户，生成链表

如果文件存在，读取文件信息，生成链表

```
int getTime(int* tm)
```

获取当前时间

```
bool nameCmp(char* newName, sudokuFile* head)
```

在链表中寻找是否有相同的名字的文档。如果没有，返回 true；如果有相同的文档，返回false

```
int add(Sudoku* s, sudokuFile* head, char* fileName, char* newName)
```

往文件中添加当前数独的信息，以保存。

```
int isSearch(Sudoku* s, sudokuFile* head, int id)
```

在链表中寻找是否有相同编号的存档。如果有，更新数独，返回 1；如果没有，返回 0；

```
int read(sudokuFile* head)
```

打印链表中的内容，包含 编号-名称-时间

```
void freeList(sudokuFile *head)
```

释放链表占用的内存

## 实验总结

本次实验利用的函数模块化的特点，使实验设计思路清晰明确；

运用了递归调用，栈函数，链表，结构体，随机数等数据结构与算法设计，对程序设计Ⅱ上课以来的知识有了更加深刻的体会

使用 <windows.h>库中的函数，对键盘的读取，输入输出光标的控制，以及命令行界面的清空，延迟等函数。对程序设计流程的控制有所体会

实验不足：

1. 未能实现图形化操作；
2. 对于递归调用解数独的过程，有时程序运行时间会很长，其算法还有很大的改进空间