



南开大学
Nankai University

计算机学院
软件工程结对编程用户手册

数独游戏程序

姓名：牟迪 宋彦艳

学号：2011360 2013471

专业：计算机科学与技术

2023 年 6 月 29 日

目录

1	文档说明	2
1.1	文档概述	2
1.2	文档适用者	2
2	程序介绍	2
2.1	游戏背景	2
2.2	核心功能	2
3	用户使用流程图	2
4	用户操作指南	3
4.1	运行环境要求	3
4.2	部署程序	3
4.3	执行参数说明	3
4.4	功能展示	3
4.4.1	c 参数——生成数独终盘	3
4.4.2	n 参数——生成数独游戏	4
4.4.3	m 参数——控制游戏难度	5
4.4.4	r 参数——控制游戏空缺范围	5
4.4.5	u 参数——生成唯一解的数独游戏	6
4.4.6	s 参数——读入游戏并求解数独	7

1 文档说明

1.1 文档概述

本文档为用户手册，撰写目的为帮助普通用户更好地了解该程序的功能，并能够合理规范地使用该程序。

1.2 文档适用者

本文档主要面向以下读者：数独游戏程序的使用用户与程序开发维护人员。

2 程序介绍

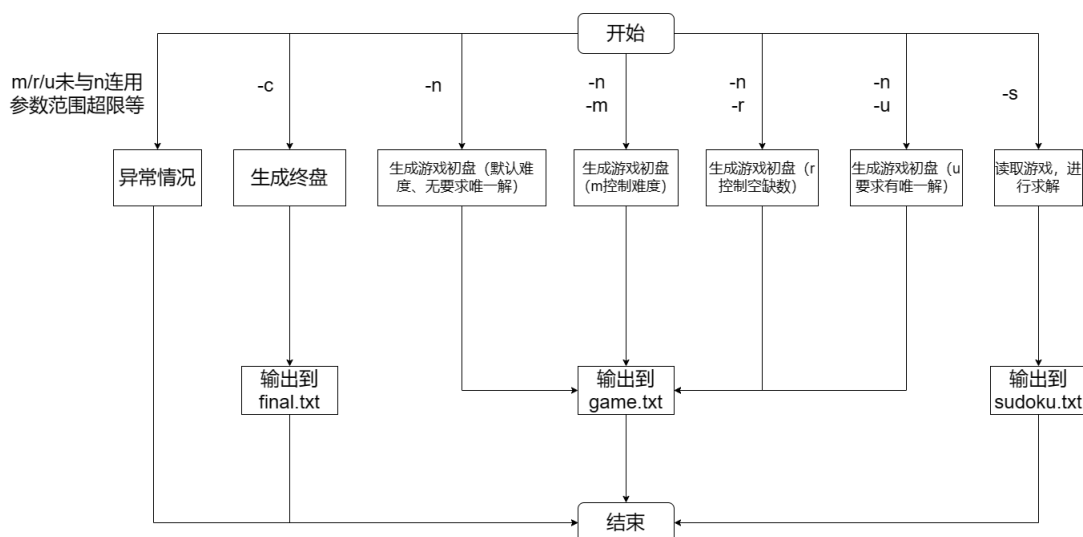
2.1 游戏背景

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，并且不能重复。

2.2 核心功能

- 生成不重复的数独终局至文件 final.txt
- 生成不重复的数独游戏至文件 game.txt
- 读取数独游戏，进行求解并将结果输出至文件 sudoku.txt
- 玩家自主控制生成的数独游戏难度
- 玩家自主控制生成的数独游戏的空缺数
- 玩家自主控制生成的数独游戏是否有唯一解

3 用户使用流程图



4 用户操作指南

4.1 运行环境要求

- 推荐操作系统：64-bit Windows 10
- 推荐 IDE：Visual Studio Code、Microsoft Visual Studio、CLion、Code::Blocks
- 编译环境要求：gcc 11.3.0 版本及以上

4.2 部署程序

下载作业提交压缩包并解压，在该可执行程序的目录下输入 cmd，进入终端。



4.3 执行参数说明

参数名字	参数意义	范围限制	用法示例
-c	需要的数独终盘数量	1-1000000	示例: sudoku.exe -c 20 [表示生成20个数独终盘]
-s	需要解的数独棋盘文件路径	绝对或相对路径	示例: sudoku.exe -s game.txt [表示从game.txt读取若干个数独游戏，并给出其解答，生成到sudoku.txt中]
-n	需要的游戏数量	1-10000	示例: sudoku.exe -n 1000 [表示生成1000个数独游戏]
-m	生成游戏的难度	1-3	示例: sudoku.exe -n 1000 -m 1 [表示生成1000个简单数独游戏，只有m和n一起使用才认为参数无误，否则请报错]
-r	生成游戏中挖空的数量范围	20-55	示例: sudoku.exe -n 20 -r 20~55 [表示生成20个挖空数在20到55之间的数独游戏，只有r和n一起使用才认为参数无误，否则请报错]
-u	生成游戏的解唯一		示例: sudoku.exe -n 20 -u [表示生成20个解唯一的数独游戏，只有u和n一起使用才认为参数无误，否则请报错]

4.4 功能展示

4.4.1 c 参数——生成数独终盘

参数名字	参数意义	范围限制	用法示例
-c	需要的数独终盘数量	1-1000000	示例: sudoku.exe -c 20 [表示生成 20 个数独终盘]

向终端输入 `sudoku.exe -c 20`：

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -c 20
c is 20
n is 0
m is 1
range is 0—0
u is 0
```

可以看到，在同目录下生成了终盘文件 final.txt：

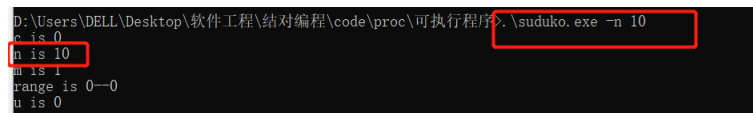


文件中生成了 20 个终盘，经检验，满足每一行、每一列、每一个九宫格内的数字均含 1-9，且不重复。

4.4.2 n 参数——生成数独游戏

参数名字	参数意义	范围限制	用法示例
-n	需要的游戏数量	1-10000	示例: suduko.exe -n 1000 [表示生成 1000 个数独游戏]

向终端输入 `suduko.exe -n 10`：



可以看到，在同目录下生成了游戏文件 game.txt：



文件中生成了 10 个初盘，经检验，满足出现数字为 1-9，且存在空缺位待求解（空缺用 \$ 表示）。注意这里默认游戏难度为 1 级，并且不要求生成唯一解的游戏。

4.4.3 m 参数——控制游戏难度

参数名字	参数意义	范围限制	用法示例
-m	生成游戏的难度	1-3	示例: sudoku.exe -n 1000 -m 1[表示生成 1000 个简单数游戏, m 需和 n 一起使用]

难度控制是由空缺数的个数决定。空缺数越多，数独求解难度越大。因此规定如下：一级难度空缺数为 27；二级难度空缺数为 36；三级难度空缺数为 45；

向终端输入 `sudoku.exe -n 10 -m 1`、`sudoku.exe -n 10 -m 2`、`sudoku.exe -n 10 -m 3`：

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -m 1
c is 0
n is 10
m is 1
range is 0--0
n is 0

D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -m 2
c is 0
n is 10
m is 2
range is 0--0
n is 0

D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -m 3
c is 0
n is 10
m is 3
range is 0--0
n is 0
```

将每次执行后，生成的不同难度游戏文件进行对比：

m1.txt - 记事本	m2.txt - 记事本	m3.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)	文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)	文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
3 8 \$ 4 \$ 6 5 7 1	9 \$ \$ 7 4 \$ 1 \$ \$	\$ \$ 9 4 \$ \$ 8 \$ 5
5 \$ 1 3 \$ 9 4 \$ 6	1 \$ 6 9 2 5 \$ \$ \$	\$ 3 5 \$ \$ 4 \$ \$
\$ 2 \$ \$ 7 \$ \$ \$ 9	7 \$ \$ \$ \$ 6 9 2 5	4 6 7 \$ \$ 5 2 \$ \$
\$ 3 8 6 4 2 \$ 5 \$	5 \$ 2 3 7 \$ \$ 1 8	\$ \$ 1 7 4 \$ 5 \$ \$
\$ 5 7 9 3 \$ 6 \$ 2	6 1 \$ \$ 9 2 3 7 4	5 8 3 \$ \$ 7 \$ 6
6 \$ \$ 1 5 7 9 3 8	\$ \$ \$ 6 1 \$ \$ 9 2	\$ 4 \$ \$ 8 3 \$ 2 1
\$ 9 \$ 2 6 4 \$ 1 5	\$ \$ \$ 4 3 \$ 8 6 \$	\$ \$ 2 6 7 \$ \$ \$ 8
7 1 5 \$ 9 3 \$ 6 \$	\$ 6 1 \$ 5 \$ 4 3 \$	\$ \$ \$ 1 9 \$ \$ 4 \$
2 6 4 7 1 \$ \$ 9 3	4 3 \$ 8 \$ \$ \$ 5 9	\$ \$ \$ 3 \$ 8 \$ 9 \$
2 1 4 8 7 \$ 5 9 6 3	\$ 6 9 \$ 7 \$ \$ \$ 8	\$ 9 3 \$ 2 \$ 4 \$ 7
9 6 \$ \$ 1 4 8 7 \$	3 4 \$ \$ \$ \$ 2 7 \$	\$ \$ 7 \$ \$ \$ \$ 2 \$
8 7 \$ \$ \$ 3 2 1 4	2 7 1 \$ 4 8 \$ \$ 9	\$ 2 8 4 \$ \$ 5 \$ \$
\$ 2 1 \$ 8 \$ \$ 9 \$	9 5 \$ \$ 2 7 8 3 4	3 \$ \$ \$ \$ \$ 4 \$
\$ 3 6 6 2 \$ 5 8 6	8 2 4 6 6 6 6 7	7 4 1 6 5 6 8 6 2
一级难度27空	二级难度36空	三级难度45空

三个文件分别生成了 10 个一级、二级、三级难度的初盘，经检验，满足一级难度有 27 个空；二级难度有 36 个空；三级难度有 45 个空。

另外注意，-m 参数的范围只能是 1 3，并且-m 参数需要和-n 一起使用才认为参数无误：

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -m 3
ERROR: argv needs -n parameter!

D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -m 40
ERROR: level should in 1'3!
```

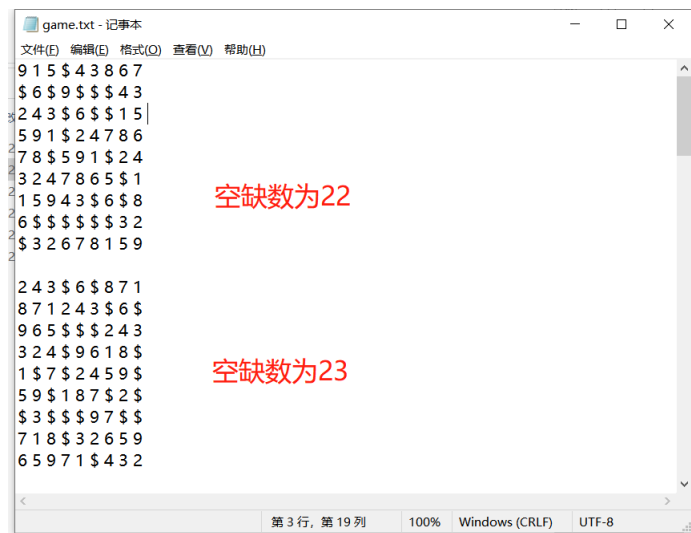
4.4.4 r 参数——控制游戏空缺范围

参数名字	参数意义	范围限制	用法示例
-r	生成游戏中挖空的数量范围	20-55	示例: sudoku.exe -n 20 -r 20 55[表示生成 20 个挖空数在 20 到 55 之间的数独游戏, r 需和 n 一起使用]

向终端输入 `sudoku.exe -n 10 -r 22~30`:

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -r 22~30
c is 0
n is 10
m is 1
range is 22~30
u is 0
```

生成的空缺数在 22~30 的游戏文件如下:



文件生成了 10 个初盘, 经检验, 它们的空缺数都在 22~30 之间。

另外注意, `-r` 参数的范围只能是 22~55, 并且 `-r` 参数需要和 `-n` 一起使用才认为参数无误:

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -r 22~30
ERROR: argv needs -n parameter!
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -r 22~60
ERROR: r1`r2 should in 20`55!
```

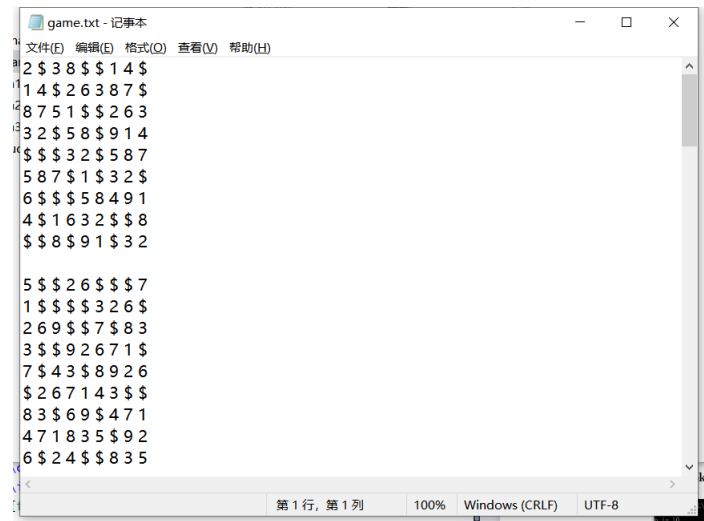
4.4.5 u 参数——生成唯一解的数独游戏

参数名字	参数意义	范围限制	用法示例
-u	生成游戏的解唯一		示例: <code>sudoku.exe -n 20 -u</code> [表示生成 20 个解唯一的数独游戏, u 需和 n 一起使用]

向终端输入 `sudoku.exe -n 10 -u`:

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序>. \sudoku.exe -n 10 -u
c is 0
n is 10
m is 1
range is 0~0
u is 1
```

生成的游戏文件如下:



文件生成了 10 个初盘，通过求解这些数独，可以验证它们的解都是唯一的。
另外注意，-u 参数需要和-n 一起使用才认为参数无误：

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序.\sudoku.exe -u
ERROR: argv needs -n parameter!
```

4.4.6 s 参数——读入游戏并求解数独

参数名字	参数意义	范围限制	用法示例
-s	需要解的数独棋盘文件路径	绝对或相对路径	示例: sudoku.exe -s game.txt [表示从 game.txt 读取若干个数独游戏，并给出其解答，生成到 sudoku.txt 中]

首先可以通过执行 `sudoku.exe -n 3` 生成游戏文件 `game.txt`，然后执行 `sudoku.exe -s game.txt`，读取文件进行求解：

```
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序.\sudoku.exe -n 3
c is 0
n is 3
m is 1
range is 0--0
u is 0
D:\Users\DELL\Desktop\软件工程\结对编程\code\proc\可执行程序.\sudoku.exe -s game.txt
```

生成的求解结果输出到 `sudoku.txt` 中，对比求解结果和原游戏文件：



sudoku.txt 文件生成了 3 个游戏初盘对应的求解结果，经过验证，求解结果是正确的。