

俄罗斯方块实验报告

Experiment Presentation of Easy Tetris

PB22111679 孙婧雯

PB22111711 陈昕琪

2022年12月25日

功能简介

Easy Tetris 程序所要达到的功能是，呈现具有三种难度模式的、简易的Tetris小游戏，游戏界面有基本操作提示和当前、历史得分记录，力求最终达到流畅易上手、游戏体验良好的效果。 游戏无BGM。

目录

- [概要设计](#)
- [详细设计](#)
- [功能展示](#)
- [存在的问题](#)
- [团队分工](#)
- [总结与建议](#)
- [参考资料](#)

1. 概要设计

1.1 使用的头文件与库函数

<stdio.h>

fclose() 用于关闭文件流，具体的文件是历史最高分记录 (.txt)。

fopen() 用于打开文件流，具体的文件是历史最高分记录 (.txt)。

fread() 用于从文件流中读取历史最高分的数据。

fwrite() 用于写历史最高分到文件流。

printf() 格式化输出函数。

scanf() 格式化输入函数。

if()-else 与 switch()-case 均为选择分支语句使用的函数。

for() 与 while() 均为循环语句使用的函数。

<Windows.h>

system("cls") 用于清屏。

system("title") 用于设置.exe会话窗口的标题

system("mode") 用于配置.exe会话窗口，具体是长x宽的尺寸。

system("pause>nul") 用于暂停程序运行，按任意键继续。

Sleep() 用于延迟程序运行。

SetConsoleTextAttribute() 用于设置控制台窗口字体颜色和背景色；使用十进制颜色对照表。

SetConsoleCursorInfo() 用于获取控制台窗口光标大小和可见性的信息。

SetConsoleCursorPosition() 用于获取控制台窗口光标位置。

GetStdHandle() 用于获得Windows标准句柄，具体的对象在本实验中是光标。

<stdlib.h>

exit() 用于终止程序运行。

rand() 用于生成随机数。

srand() 用于初始化随机种子（发生器）。

此头文件中也含有system()函数。

<time.h>

time() 用于获取系统时间，常见于随机种子生成器。

<conio.h>

getch() 用于无回显地从控制台输入一个字符。

kbhit() 用于检查键盘是否有输入，有返回-1，无返回0。

十进制颜色对照表															
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

十进制颜色对照表

1.2 定义的函数与调用关系

代码中所有自定义的函数如下：

```
void HideCursor();           //用于隐藏光标
void Gotoxy(int x, int y);   //用于使光标跳转到指定的(x,y)位置
void InitPlayArea();        //用于初始化游戏区·绘制边框·给出游戏说明
void InitBlock();           //用于初始化方块形态
int ShowMenu();             //用于显示初始菜单并选择模式
void ClearArea();           //用于游戏开始前清除初始菜单
void color(int num);        //用于设置控制台窗口字体颜色·使用十进制颜色对照表
void DrawBlock(int shape, int form, int x, int y); //用于画出方块
void EraseBlock(int shape, int form, int x, int y); //用于擦除方块
int IsLegal(int shape, int form, int x, int y);    //用于合法性判断(能否移动/旋转)
int IsCount();             //用于计算得分
void Congratulations();     //得分后的小动画
void IsGameOver();          //用于判断是否gameover
void StartGame();           // *此为游戏的主函数
void SPause();              //用于暂停游戏
void ReadRecord();          //用于从文件读取最高分
void WriteRecord();         //用于更新最高分到文件
int main()                  //main函数
```

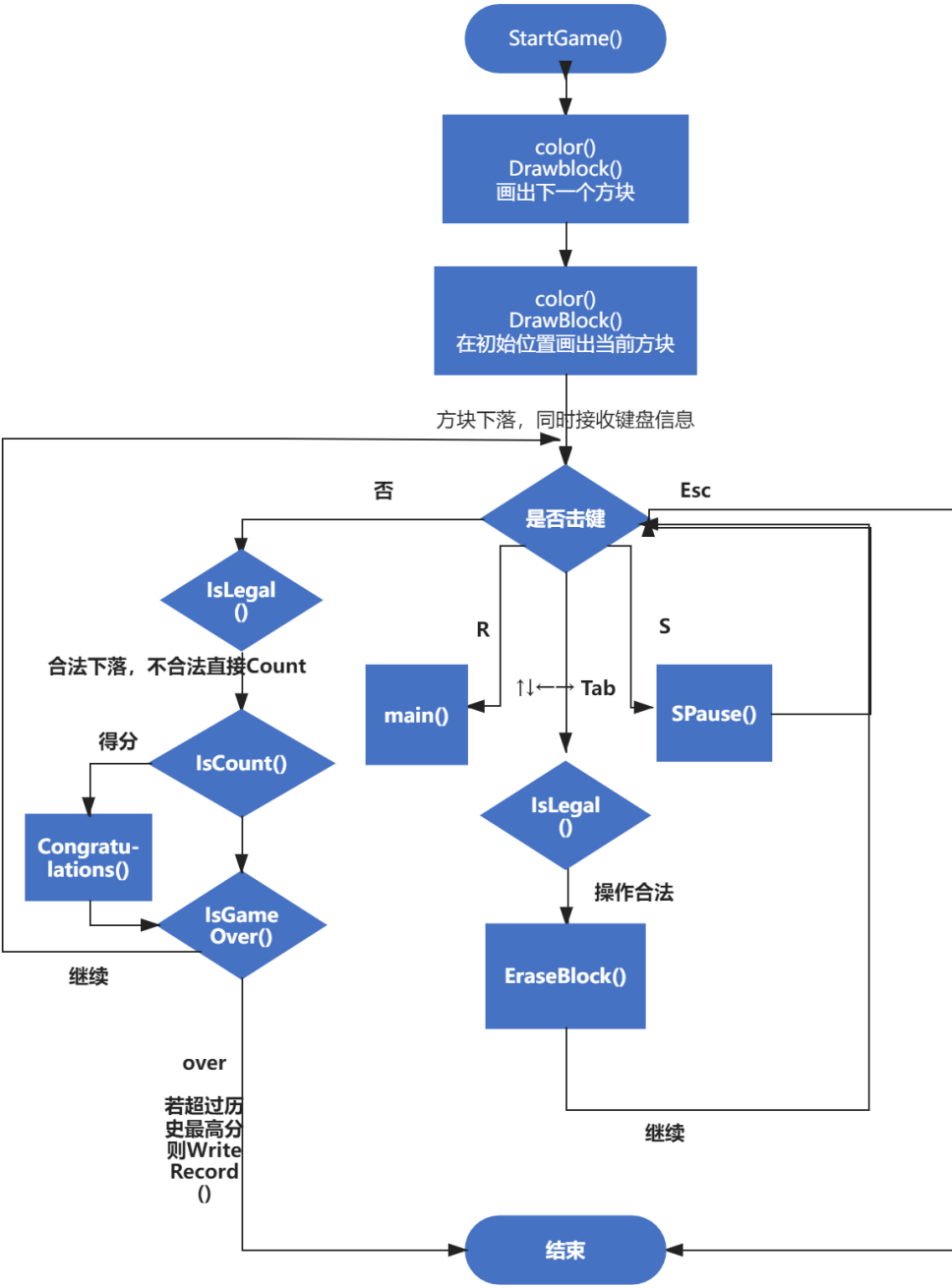
程序运行时，首先调用main()函数，main()函数内部依次调用HideCursor()隐藏光标、ShowMenu()选择游戏模式、ReadRecord()加载历史分数、InitPalyArea()加载游戏区、InitBlock()加载方块形态、StartGame()开始游戏。

ShowMenu()函数中，调用Gotoxy()函数在指定位置写出文字，玩家选择游戏模式之后会调用ClearArea()函数清屏。

InitPlayArea()函数中，调用Gotoxy()函数在指定位置画出游戏区边框、写出文字提示游戏玩法。

StartGame()函数中，主体框架是两层while()循环，外层循环控制每一个方块下落全程并提示玩家下一个方块的形态，内层循环控制每一个方块下落一格并接收键盘上的移动、旋转、暂停等操作。顺序执行时，程序首先进入第1层循环，调用color()和DrawBlock()画出下一个方块，再进入第2层循环，调用DrawBlock()画出当前方块，方块根据玩家选择的模式以不同速度下落。每下落一格接收一次键盘上是否有操作，若有且为移动/旋转操作，调用IsLegal()判断是否合法操作，合法的则调用EraseBlock()擦除当前位置显示的方块，预备进入下一次内层循环时在新的位置画出方块；若有且为暂停操作，调用SPause()暂停；若有且为退出操作，终止游戏；若有且为重开操作，直接调用main()重新加载游戏。接收结束后若游戏继续，会重新进入内层循环。若没有接收到键盘操作，调用IsLegal()、IsCount()判断是否继续向下移动、是否得分。IsCount()中，若得分会调用Congratulations()出现加分小动画。在IsCount()的结尾，调用IsGameOver()判断是否结束游戏。每次结束游戏时，若打破最高分记录，则调用WriteRecord()记录新高分。

因此StartGame()的执行流程图如下：



经测试，一旦开始执行`StartGame()`函数，玩家可以随时进行键盘操作，且gameover和暂停均有反应时间，可以获得较良好的游戏体验。

控制键键码值 (keyCode)								字母和数字键的键码值 (keyCode)							
按键	键码	按键	键码	按键	键码	按键	键码	按键	键码	按键	键码	按键	键码	按键	键码
BackSpace	8	Esc	27	Right Arrow	39	-_	189	A	65	J	74	S	83	1	49
Tab	9	Spacebar	32	Dw Arrow	40	.>	190	B	66	K	75	T	84	2	50
Clear	12	Page Up	33	Insert	45	/?	191	C	67	L	76	U	85	3	51
Enter	13	Page Down	34	Delete	46	~	192	D	68	M	77	V	86	4	52
Shift	16	End	35	Num Lock	144	[{	219	E	69	N	78	W	87	5	53
Control	17	Home	36	::	186	\	220	F	70	O	79	X	88	6	54
Alt	18	Left Arrow	37	=+	187	}]	221	G	71	P	80	Y	89	7	55
Cape Lock	20	Up Arrow	38	,<	188	""	222	H	72	Q	81	Z	90	8	56
								I	73	R	82	0	48	9	57

StartGame()中接收键盘信息使用的键码值表

详细设计

(贴代码)

功能展示

(贴截图)

存在的问题

(没BGM)(不会延时下落)

团队分工

陈昕琪 xxxxxxxx

孙婧雯 XXXX

总结和建议

(助教浇浇)

参考资料

(贴网址)