color(7); //颜色设置为白色

CursorJump(0, ROW);

printf("当前得分:%d", grade);

CursorJump(COL, ROW);

printf("历史最高得分:%d", max);

}

//颜色设置

void color(int c)

{

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), c); //颜色设置

//注：SetConsoleTextAttribute是一个API（应用程序编程接口）

}

//从文件读取最高分

void ReadGrade()

{

FILE\* pf = fopen("贪吃蛇最高得分记录.txt", "r"); //以只读的方式打开文件

if (pf == NULL) //打开文件失败

{

pf = fopen("贪吃蛇最高得分记录.txt","w"); //以只写的方式打开文件

fwrite(&max, sizeof(int), 1, pf); //将max写入文件（此时max为0），即将最高得分初始化为0

}

fseek(pf, 0, SEEK\_SET); //使文件指针pf指向文件开头

fread(&max, sizeof(int), 1, pf); //读取文件当中的最高得分到max当中

fclose(pf); //关闭文件

pf = NULL; //文件指针及时置空

}

//更新最高分到文件

void WriteGrade()

{

FILE\* pf = fopen("贪吃蛇最高得分记录.txt", "w"); //以只写的方式打开文件

if (pf == NULL) //打开文件失败

{

printf("保存最高得分记录失败\n");

exit(0);

}

fwrite(&grade, sizeof(int), 1, pf); //将本局游戏得分写入文件当中

fclose(pf); //关闭文件

pf = NULL; //文件指针及时置空

}

//初始化蛇

void InitSnake()

{

snake.len = 2; //蛇的身体长度初始化为2

snake.x = COL / 2; //蛇头位置的横坐标

snake.y = ROW / 2; //蛇头位置的纵坐标

//蛇身坐标的初始化

body[0].x = COL / 2 - 1;

body[0].y = ROW / 2;

body[1].x = COL / 2 - 2;

body[1].y = ROW / 2;

//将蛇头和蛇身位置进行标记

face[snake.y][snake.x] = HEAD;

face[body[0].y][body[0].x] = BODY;

face[body[1].y][body[1].x] = BODY;

}