**概要设计说明书**

**《西柚贪吃蛇》**

**编写日期：2019/11/03**

项目组成员：

|  |  |  |
| --- | --- | --- |
| 学号 | 姓名 | 角色 |
| 201731024203 | 黄耀萱 | 查询相关代码、系统设计说明书 |
| 201731024101 | 李楠 | 编纂系统设计博客、演讲PPT |
| 201731022104 | 张莹 | 做需求分析PPT、需求规格说明书 |
| 201731024201 | 孙颖 | 查询相关代码、调试代码 |
| 201731024105 | 汪小萍 | 做系统设计PPT |
| 201731131317 | 杨也 | 编纂需求说明博客 |

**修改日志**

|  |  |  |
| --- | --- | --- |
| 修改者 | 修改日期 | 备注说明 |
| 黄耀萱 | 2019/10/13 | 编写软件系统结构设计相关内容 |
| 黄耀萱 | 2019/10/15 | 修改接口设计 |
| 黄耀萱 | 2019/11/03 | 软件概念设计 |
| 孙颖 | 2019/11/03 | 功能需求与程序的关系 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**目 录**

[1. 引言 5](#_Toc23704805)

[1.1 编写目的 5](#_Toc23704806)

[1.2 定义 5](#_Toc23704807)

[1.3 参考资料 5](#_Toc23704808)

[2. 范围 5](#_Toc23704809)

[2.1 系统主要目标 5](#_Toc23704810)

[2.2 主要软件需求 6](#_Toc23704811)

[2.3 设计约束、限制 6](#_Toc23704812)

[3. 软件概念及系统结构设计 7](#_Toc23704813)

[3.1 软件概念设计 7](#_Toc23704814)

[3.2 软件体系结构 7](#_Toc23704815)

[3.2.1 软件程序结构图 7](#_Toc23704816)

[3.2.2 模块描述 7](#_Toc23704817)

[3.3 功能需求追溯 8](#_Toc23704818)

[4. 接口设计 9](#_Toc23704819)

[4.1 用户界面设计规则 9](#_Toc23704820)

[4.2 内部接口设计 9](#_Toc23704821)

[4.3 外部接口设计 9](#_Toc23704822)

[5. 功能需求与程序的关系 9](#_Toc23704823)

[6. 出错处理设计 9](#_Toc23704824)

# 引言

## 编写目的

在分析《贪吃蛇游戏的需求分析说明》的基础上，我们对该系统做了概要设计，主要是基于以下目的编写此说明书：

1、对系统概要设计的的阶段任务成果形成文档，以便阶段验收、评审，最终的阶段验收。

2、对需求阶段的文档再次确认过程，对前一阶段的需求不充分的地方或者有错误的地方进行修改。

3、明确整个系统的功能框架和数据结构，为下一阶段的详细设计、编码、测试提供参考依据。

4、明确编码规范和命名规范，统一程序界面。

## 定义

1. **节：**

一条蛇可以看成由许多正方形的“小格子”拼凑成，称作节。节是蛇身上最小的单位。

1. **链表：**

用来保存每一段的状态，链表的元素单位是节。且链表的第一个元素表示蛇的头部。

**3、坐标系：**

以左上角那点为（0,0），向右则x递增，向下则y递增。

## 参考资料

张海藩：《软件工程导论》第五版清华大学出版社

肖刚等：《实用软件文档写作》

# 范围

## 系统主要目标

本设计所开发的是基于C#的一个贪吃蛇游戏软件，主要实现游戏的开始、暂停、退出等功能，分为：游戏主界面模块、游戏控制模块两个模块。

本游戏在设计方面本着方便、实用及娱乐性高的宗旨，在对界面进行设计的过程中，始终坚持清晰明了，在性能方面能够实现效率高，不易出错等优点。游戏主界面应该力求美观，赏心悦目。游戏控制模块应做到易懂、易操作，而且准确率高，不易出错。

## 主要软件需求

**（一）系统质量需求**

1、**性能**

（1）**系统容量：**本软件由于过于简单，在系统容量上没有太大需求。一般电脑均可使用。

（2）**响应指标：**进入游戏，点击“开始”后，“蛇”在0.01秒内做出反应，开始移动；在“吃到食物”以及“上”和“下”等各项命令指标是反应时间均在0.01秒内。

2、**可维护性：**在可维护性方面，用户可联系技术人员，由他们亲自来解决维护问题。

3、**可用性：**本软件无需联网，支持脱机运行。因此，用户可以随时随地使用本软件。

4、**可移植性:**本软件可看成独立系统，所以实现了U盘的拷贝、粘贴等行为。

5、**可测试性：**作为一款游戏类项目，在测试方面用了大量的时间，而这项目的克测试性非常的强，因为游戏本身也算一种测试。

6、**易用性：**软件所占容量小，游戏本身操作简单，更人性化的界面，上手极快。

**（二）安全性需求**

1、**网络安全：**由于本软件不需要联网，所以网络安全这一问题可轻松避免。

2、**数据库安全：**数据库级备份和恢复，数据库级用户进行角色和权限授权。使在异常情况发生时，系统能够得以快速恢复，避免数据的丢失或将其影响降到最低。同样，要保证存储过程中不被非法的访问和篡改。

3、**应用系统的安全：**通过对用户的身份鉴别，使用户只能得到系统授权的数据访问功能操作。用户只能在授权下才可以更改程序，避免因程序更改而引起程序出错。

## 设计约束、限制

由于本项目开发人员皆为初学者，所有在各个方面都略显不足，应该会有不少错误和漏洞。此外，在系统方面，我们也是采用较为简单的表达，并不是十分完美。

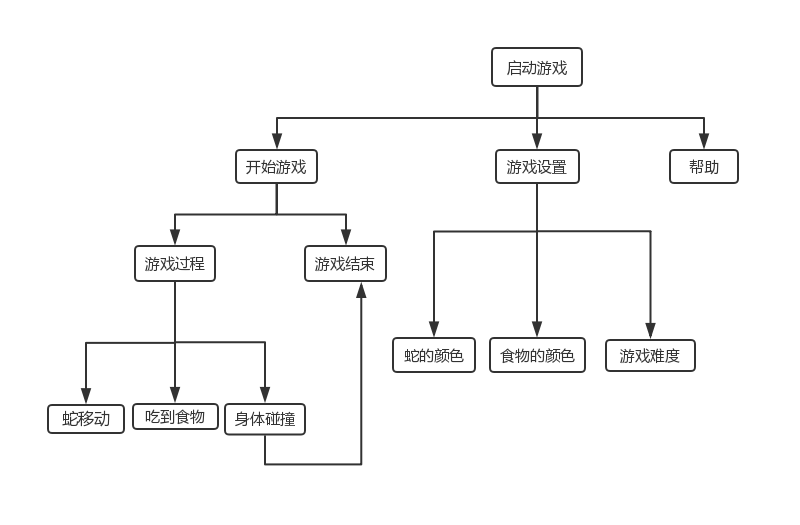
# 软件概念及系统结构设计

## 软件概念设计

程序关键在于表示蛇的图形及蛇的移动。用一个小矩形快表示蛇的一节身体，身体每长一节，在身体最后增加一个矩形块。移动时必须从蛇头开始，所以蛇不能向相反的方向移动，如果不按任意键，蛇自行在当前方向上前移，但按下有效方向键后，蛇头朝着该方向移动，一步移动一节身体，所以按下有效方向键后，先确定蛇头的位置，而后蛇的身体随蛇头移动，图形的实现是从蛇头新位置开始画出蛇，长度为保存蛇每个节点信息的链表的长度。食物的出现是先确定一组随机数，再填充方格，食物的消失是由于使用了清屏函数，当吃掉食物后就不再重画本次食物，而是生成一组新的随机数，画新的食物。当蛇头的位置坐标和石头的坐标或者自己身体的某个节点的坐标相同时，蛇就死掉了，游戏结束。

## 软件体系结构

### **软件程序结构图**



### **模块描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模 块名 称 | 游戏设置 | | 子系统名称 | 蛇的颜色、食物的颜色，游戏难度 |
| 接　口说　明 | 输入 | 点击设置的相关按钮 | | |
| 输出 | 蛇和食物的颜色以及游戏难度发生相应的改变 | | |
| 功　能说　明 | 玩家可通过设置按钮改变蛇的颜色，食物的颜色和游戏难度。 | | | |
| 运行环境说明 | VS2015；编程语言:C# | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模 块名 称 | 游戏过程 | | 子系统名 称 | 蛇移动 |
| 接　口说　明 | 输入 | “上”“下”“左”“右”控制键 | | |
| 输出 | 蛇分别向“上”“下”“左”“右”移动 | | |
| 功　能说　明 | 玩家通过‘上、下、左、右’来控制目标的行进方向,而目标会做出相应的反应。 | | | |
| 运行环境说明 | VS2015；编程语言:C# | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模 块名 称 | 游戏过程 | | 子系统名 称 | 吃到食物 |
| 接　口说　明 | 输入 | 通过控制键操控蛇身碰到食物 | | |
| 输出 | 食物消失，蛇身增长一格 | | |
| 功　能说　明 | 玩家操控蛇吃到食物可以使蛇获得蛇身增长的加成。 | | | |
| 运行环境说明 | VS2015；编程语言:C# | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模 块名 称 | 游戏过程 | | 子系统名称 | 身体碰撞 |
| 接　口说　明 | 输入 | 通过控制键操控蛇碰到蛇身或墙壁 | | |
| 输出 | 游戏结束 | | |
| 功　能说　明 | 玩家操控蛇碰到蛇身或墙壁就算游戏失败，游戏结束。 | | | |
| 运行环境说明 | VS2015；编程语言:C# | | | |

## 功能需求追溯

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 设置 | 蛇移动 | 吃到食物 | 身体碰撞 |
| 改变蛇的颜色 | **√** |  |  |  |
| 改变食物颜色 | **√** |  |  |  |
| 选择游戏难度 | **√** |  |  |  |
| 蛇开始行进 |  | **√** |  |  |
| 蛇身增长 |  |  | **√** |  |
| 游戏失败 |  |  |  | **√** |

# 接口设计

## 用户界面设计规则

1、**用户界面：**

(1)要求有菜单及工具栏以方便操作。

(2)进行系统分析后的结果在另一窗口中显示。

2、**界面风格：**朴素、简单；清晰、明朗。

3、**界面规范：**游戏所需的所有选项均在界面，使其显示清晰明了、游戏简单易上手。

## 内部接口设计

SnakeListener：

Void SankeMoved（Snake）；

## 外部接口设计

Windows标准接口

# 功能需求与程序的关系

1、游戏界面主框架主要包括游戏图形主界面、开始界面。

1)游戏图形主界面主要包括:食物的随机位置、石头的布局、蛇的移动位置

2)开始 界面包括:开始、继续、暂停、历史最高分、游戏说明、退出等按钮。

2、与程序的关系:

1)开始界面StartPanel画出开始、继续、暂停、历史最高分、游戏说明等按钮。

2)游戏图形 主界面GamePanel画出Snake, Ground, Food。 用户在StartPanel选择按钮或者按下快捷键，Controller 类判断逻辑或响应事件，向Snake,Ground, Food 发出命令，分别执行，最后由GamePanel实现刷新并显示。

# 出错处理设计

|  |  |  |
| --- | --- | --- |
| 可能的出错或故障情况 | 按钮连续选择两次 | 游戏在进行时选择新游戏 |
| 系统输出信息的形式、含意 | 蛇的速度加快或是抛出异常 | 蛇的速度加快 |
| 处理方法 | 每个按钮在选择后设为灰显 | 在游戏开始后newGame按钮变灰显 |