\*\*\*\*

需求分析与概要设计说明书

|  |  |
| --- | --- |
| 学号 | 15281206 |
| 姓名 | 郭梦霏 |
| 班级 | 1508 |
| 电话 | 18810467595 |
| Email | [15281206@bjtu.edu.cn](mailto:15281206@bjtu.edu.cn) |
| 其它联系方式 |  |
| 版本号 |  |
| 最修改日期 | 2016.6.12 |

目 录

[1．引言 2](#_Toc260123422)

[1.1编写目的 2](#_Toc260123423)

[1.2项目背景 2](#_Toc260123424)

[1.3定义 2](#_Toc260123425)

[1.4参考资料 2](#_Toc260123426)

[2．任务概述 2](#_Toc260123427)

[2.1系统总体目标 2](#_Toc260123428)

[2.2运行环境需求 2](#_Toc260123429)

[3．数据需求与概要设计 2](#_Toc260123430)

[3.1数据描述 2](#_Toc260123431)

[3.2数据字典 2](#_Toc260123432)

[3.3数据内外部存储 2](#_Toc260123433)

[3.4数据流说明 2](#_Toc260123434)

[4．功能需求与概要设计 2](#_Toc260123435)

[4.1功能划分 2](#_Toc260123436)

[4.2功能描述 2](#_Toc260123437)

[5．性能需求 2](#_Toc260123438)

[5.1数据精确度 2](#_Toc260123439)

[5.2时间特性 2](#_Toc260123440)

[5.3适应性 2](#_Toc260123441)

[6．运行需求 2](#_Toc260123442)

[6.1用户界面 2](#_Toc260123443)

[6.2硬件接口 2](#_Toc260123444)

[6.3软件接口 2](#_Toc260123445)

[6.4故障处理需求 2](#_Toc260123446)

[7．其它需求 2](#_Toc260123447)

# 1．引言

## 1.1编写目的

为明确软件需求、安排项目规划与进度、组织软件开发与测试，撰写本文档。

本文档供项目经理、设计人员、开发人员参考。

扫雷是以C++语言为基础的应用系统。本系统包含运行游戏、设置游戏规格、高分榜、计时器等功能。

## 1.2项目背景

1. 项目的委托单位、开发单位和主管部门

委托单位：北京交通大学计算机学院 开发者：郭梦霏

1. 该软件系统与其他 项目完成时间：8周

## 1.3定义

无

## 1.4参考资料

1. 项目经核准的计划任务书、合同或上级机关的批文
2. 项目开发计划
3. 文档所引用的资料、标准和规范。列出这些资料的作者、标题、编号、发表日期、出版单位或资料来源

# 2．任务概述

## 2.1系统总体目标

本项目的目的在于开发一个扫雷软件，该软件面向游戏玩家用户，具有进行扫雷游戏的功能。

与系统的相关的用户包括游戏玩家，用户与本系统的关系如下：

游戏玩家将使用系统中的如下功能：

1）进行游戏；2）使用自定义游戏规格功能；3）使用计时器功能；4）使用高分榜功能。

## 2.2运行环境需求

本软件可以运行在普通微机上。

# 3．数据需求与概要设计

## 3.1数据描述

本系统涉及的对象包括方格中是否有方格状态(是否有雷)（fangge[X][Y]），产雷(lei[k])，方格是否打开(Click[X][Y])，方格内数字(number[X][Y])，是否被红旗标记(Flag[X][Y])，是否被左右键点下(LeftClick[X][Y]、RightClick[X][Y])，周围红旗标记数(flagnumber[X][Y])，游戏是否胜利(win\_or\_lose)等。(X、Y代表方格坐标)

各类对象间的关系为：

对于特定的方格：

1. 方格附近的雷数决定方格显示的数字是多少；
2. 方格是否被打开（方格状态）决定是否显示雷或数字；
3. 左右键点下同时标记数等于雷数以后打开未被标记的方格；
4. 方格中有雷被打开为游戏失败；
5. 雷是否全被找出（雷状态）决定游戏是否胜利。

然后给出每类对象的详细属性列表，如：

雷：位置(随机函数布雷)(lei[k]=0~X\*Y)

方格是否有雷：有/无(fangge[X][Y]=1/0)

方格状态：点开/未被点开(Click[X][Y]=1/0)

红旗：被标记/未被标记(flag[X][Y]=1/0)

方格内数字：0~8不等(number[X][Y]=0~8)

雷状态：被排完/未被点到/被点到(win\_or\_lose=1/0/-1)

周围红旗数：0~8不等(flagnumber[X][Y]=0~8)

## 3.2数据字典

本系统将游戏数据分别装在View类和其他类中描述，游戏所具有的数据项的详细定义见表X。

表x，游戏数据的各个数据项（属性）字典（定义于View中）

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 数据项 | 名称 | 类型 | 长度精度 | 是否为关键字 | 能否为空 | 编码规划或定义域说明 | 备注 |
| fangge[50][50] |  | 整数数组 | 50\*50 |  | NO | 有：1 无：0 | 记录是否有雷 |
| Click[50][50] |  | 整数数组 | 50\*50 |  | NO | 点开：1 无：0 | 记录是否被点开 |
| number[50][50] |  | 整数数组 | 50\*50 |  | NO | 0~8 | 记录方框内数字 |
| flag[50][50] |  | 整数数组 | 50\*50 |  | NO | 是：1 否：0 | 记录是否被红旗标记 |
| LeftClick[50][50] |  | 整数数组 | 50\*50 |  | NO | 是：1 否：0 | 记录左键是否点下 |
| RightClick[50][50]  flagnumber[50][50] |  | 整数数组  整数数组 | 50\*50  50\*50 |  | NO  NO | 是：1 否：0  0~8 | 记录右键是否点下  记录周围红旗数 |
| i  j  k | 坐标  坐标  雷 | 整数  整数  整数 | 1  1  1 |  | NO  NO  NO | 0~X  0~Y  0~雷数 | 横坐标  纵坐标  雷循环变量(布雷) |
| lei[50] |  | 整数数组 | 50 |  | NO |  | 产雷上限50 |
| X |  | 整数 | 1 |  | NO | 0~50 | 游戏界面长 |
| Y |  | 整数 | 1 |  | NO | 0~50 | 游戏界面宽 |
| Mine\_number |  | 整数 | 1 |  | NO | 0~50 | 雷数 |
| win\_or\_lose |  | 整数 | 1 |  | NO | 胜：1 未结束：0  败：-1 | 是否胜利 |
| Time  second | 时间 | 整数  整数 | 1  1 |  | NO  NO | 未：0 开始：1  0~游戏时间 | 开始计时标志  计时 |
| a  \_x  \_y  set\_mine | 规格 | 对象  整数  整数  整数 | 1  1  1 |  | NO  NO  NO  NO | 通过对话框控制游戏规格  0~最大规格  0~最大规格  已进行布雷：1  未进行布雷：0 | 全局变量（ GAMESIZE）  记录鼠标点击横方格  记录鼠标点击纵方格  是否布雷 |
| win | 胜利 | 整数 | 1 |  | NO | 胜利：1 未胜利：0 | 全局变量记录是否胜利 |
| game\_time | 时间 | 整数 | 1 |  | NO | 值：second | 全局变量 记录游戏时间 |

## 3.3数据内外部存储

3.3.1 游戏数据的内外存存储

游戏数据在内存中以数组形式保存，

游戏玩家姓名数据在外存中保存成数据文件，文件格式定义如下：highscore.txt

## 3.4数据流说明

在此处说明各类数据的大致处理流程，可以用数据流图或文字进行说明，数据流图的画法请自己参考相关文档。

高分榜处理流程：

1. 由用户输入名字，存储到相应文件中
2. 用户打开高分榜
3. 打开相应文件，进行读取和显示

# 4．功能需求与概要设计

总体功能：进行扫雷游戏

## 4.1功能划分

本软件的功能划分成如下4个模块

设置游戏规格

记录玩家姓名

2) 进行扫雷游戏

3) 游戏内计时器

4) 玩家高分榜

## 4.2功能描述

4.2.1设置游戏规格

进入游戏前，弹出对话框，进行设置游戏界面规格，可设置游戏区域长、宽、雷数。

实现过程：

定义全局变量，在游戏开始后输入行数、列数、雷数，传递给CSaoLeiView进行变量初始化(初始化为10/10/10)。

4.2.2记录玩家姓名(SetName)

弹出对话框，要求玩家输入玩家姓名，便于之后记录游戏数据(高分榜)。若游戏胜利且时间少于高分榜上玩家，则添加游戏记录。

实现过程：

定义全局变量，在游戏开始后储存玩家姓名(m\_name)，之后等待是否需要被传递。

4.2.3进行扫雷游戏(CSaoLeiView)

左键单击界面开始游戏，若为空格（实际为数字0）则展开至周围全为数字，数字则显示数字，右键可标记雷，左右键同时点击可进行排雷（周围红旗数等于雷数使打开未被标记的方格）。全部雷被排以后显示游戏胜利(笑脸)，有雷被点开显示游戏失败(哭脸)。

实现过程：

对于单个的方格显示：

首先判断是否被点开，若被点开则判断是否有红旗标记，若有标记显示红旗，无则显示未点开；若被点开，则判断该方格是否有雷，若有则显示雷，同时游戏失败，显示哭脸。无则打开方格，显示数字；若数字为0，则打开附近的方格；若游戏进行到所有雷都被排出，或者只剩下雷未被点开，则游戏胜利，显示笑脸。

布雷（SetMine()）：

调用随机函数（非伪随机）产生0~X\*Y的数字，分别放到相应的方格中（布雷），若重复，则再次布雷。

展开（SetClick()）：

判断方格是否为空（0）且处于打开状态，则打开周围八格；之后遍历每格，进行打开方格。

计算数字（SetNumber()）：

通过对方格不同位置的讨论，如顶点，边，中心，进行方格数字的计算。

计算红旗（SetFlagNumber()）：

基本计算方法同上，计算周围红旗的数目。

右键标记：

由右键函数对相应数组进行操作，点下后该位置记为1，未点下则记为0。

左右键排雷：

由左键函数和右键函数对相应数组进行操作，当两数组同时为1则进行判断：周围红旗数是否等于雷数，若相等则打开未被标记的方格。

判断游戏结果：

仅剩所有雷未被点开，即可获得游戏胜利。

构造函数：

初始化各类变量（基本为0），接受设置窗口传来的游戏规格变量。

OnCreate()：

设立计时器SetTimer()。

OnDraw()：

首先进行布雷，计算方格中的数字，计算周围的红旗数（用于左右键），点击结算操作。之后输出方格、游戏界面（显示方法见上）、计时器（分三数字展示），判断是否胜利（游戏结束后关闭计时器）。

OnIDGaofen()：

将高分榜添加到游戏界面上（定义高分榜的全局变量，将参数值传递过来）。

OnLButtonUp()：

点下后开始计时，若游戏未结束，对点击位置进行计算，判断位于哪个方格，记录点击，调用OnDraw()。

OnRButtonUp()：

同上（添加再次点击取消小红旗功能）。

OnTimer()：

与之前设立的计时器相连，游戏开始后开始计时，每1s重画一次计时器区域。

4.2.4游戏内计时器

左键点击游戏界面以后开始计时，游戏结束以后停止计时。

实现过程：

之前设置SetTimer函数，在OnTimer中进行计时操作。首先判断游戏是否开始（由单一变量Time记录），左键点下以后开始计时，并以三位数的形式显示在游戏区域左上方；之后判断游戏是否结束（win\_or\_lose），结束后停止计时，储存在全局变量中，等待是否传递给高分榜。

4.2.5玩家高分榜

游戏胜利后，若游戏时间比之前高分榜上游戏玩家用时少，则记录新的游戏记录和玩家，下次打开游戏使用高分榜功能时也能够看到本次记录。高分榜记录前三名用时最少的历史玩家。

实现过程：

开启一个字符数组用于存放用户名，整数数组用于存放游戏时间（长度均为3，记录前三名玩家）。使用全局变量，传递游戏时间和用户名；定义文件指针进行读写。按一定格式对文件内容进行读取，调用Sort函数，进行排序；输出排序后的前三名到高分榜对话中，同时输出到文件中，覆盖之前的文件内容。

Sort()：

按时间进行排序，用时少的排在前面。

# 5．性能需求

## 5.1数据精确度

本程序数据均为数字和字符，精确度高。

## 5.2时间特性

无特定需求

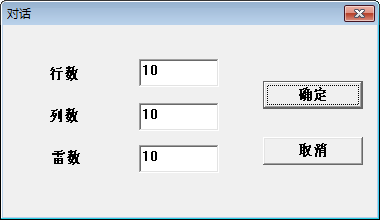
## 5.3适应性

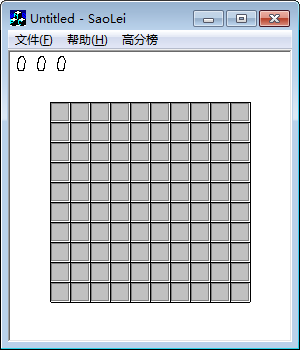
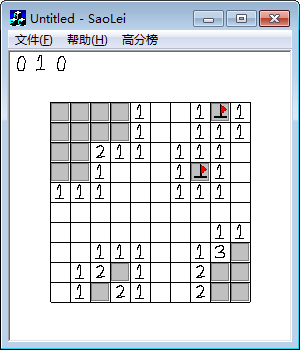
尚未测试

# 6．运行需求

## 6.1用户界面

界面截屏如下：

菜单：高分榜入口

## 6.2硬件接口

暂无。本程序为完整可单独运行游戏，无相应硬件接口。

## 6.3软件接口

暂无。本程序为完整可单独运行游戏，无相应软件接口。

## 6.4故障处理需求

本软件为一次性运行软件，游戏胜利后即可记录数据，若机器出现故障文档未保存，则无法即时记录游戏数据。

# 7．其它需求

可使用性：本程序面向全体玩家，可使用性较高。

可移植性：若要正常使用高分榜功能，移植时应当连同记录高分榜的txt文件(highscore.txt)一同移植，否则程序在使用高分榜功能时会产生错误。