**详细设计说明书**

**《黄金点游戏系统》**

**编写日期：2019年10月28日**

**项目组：**

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **角色** |
| **201831061408** | **徐鹏** | **组长** |
| **201831061328** | **向天强** | **组员** |
| **201831061329** | **肖子轩** | **组员** |
| **201831061410** | **延昌磊** | **组员** |
|  |  |  |
|  |  |  |

**修改日志**

|  |  |  |
| --- | --- | --- |
| 修改者 | 修改日期 | 备注说明 |
| 延昌磊 | 10月27日 | 建立文档 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 引言

## 编写目的

当今社会碎片化时间增多。聚会时低头现象日益严重导致许多聚会效果不佳。未达到充分利用碎片化时间和增加聚会游戏氛围，故我们反恐精英小组决定做一个简单的游戏，黄金点游戏。

## 项目背景

说明书编制的目的是说明一个软件系统各个层次中，每个程序每个模块豁子程序和数据库系统的设计考虑为程序员编码提供依据。编写详细设计说明书是软件开发过程必不可少的部分，其目的是为了使开发人员在完成概要设计说明书的基础上，完成概要设计规定的各项模块的具体实现的设计工作，同时也是开发人员和最终客户进行需求交流的有效手段。如果一个软件系统比较简单，层次，很少本文件可以不单独编写和概要设计说明书中不重复部分合并编写。

## 定义

### 引用名词1 名词1的解释

… （定义本详细设计说明书所引用的名词的含义。）

## 参考资料

### 《文档名称1》 版本号，作者，编写时间

<https://wenku.baidu.com/view/1ad0617ddd88d0d232d46a21.html>

<https://wenku.baidu.com/view/22506d58ae1ffc4ffe4733687e21af45b307fecd.html>

# 软件结构概述

模块1：选择规则，是否自定义规则，输入数字

模块2：算出平均值并的出G值

模块3：增加减少分数

模块4：增加减少分数，生成奖惩措施

# 模块设计

## 模块1

（模块以概要设计中划分的模块或类为基本单元。）

### 功能描述

选择规则，是否自定义规则，输入数字 选择人数

### 接口描述

输入 参与人数

输出 参与人数的数组

### 内部元素结构

（准确地描述每个模块包含的数据、子程序等。）

void goldPoint(){

int num;

int time;

double sum=0;

double aver;

int i,j;

int m=0;//用来初始化数组下标

scoreNode \*head,\*p,\*q,\*head0,\*q0,\*p0;

double numArray[10]={0};

head=NULL;

head0=NULL;

q=NULL;

q=head;

q0=head0;

printf("请输入参加的人数：\n");

scanf\_s("%d",&num);

printf("请输入需要进行几轮：\n");

scanf\_s("%d",&time);

printf("第1轮开始\n");

for(i=0;i<num;i++){

p=(scoreNode \*)malloc(sizeof(scoreNode));

p0=(scoreNode \*)malloc(sizeof(scoreNode));

printf("请输入你的名字：\n");

scanf\_s("%s",p->name,sizeof(p->name));

printf("请输入你的数：\n");

scanf\_s("%lf",&p->score,sizeof(p->score));<br>　　　　　　　　　　//每轮输入的数字不能相同

while(isEqual(numArray,p->score)){

printf("已经被人输入，请重新输入：\n");

scanf\_s("%lf",&p->score,sizeof(p->score));

}

numArray[i]=p->score;

p0->next=NULL;

p->next=NULL;

if(head==NULL){

head=p;

q=p;

}else{

q->next=p;

q=p;

}

sum+=p->score;

strcpy\_s(p0->name,strlen(p->name)+1,p->name);

p0->score=0;

if(head0==NULL){

head0=p0;

q0=p0;

}else{

q0->next=p0;

q0=p0;

}

}

aver=sum/num\*0.618;

//printf("aver=%lf\n",aver);

q=head;

q0=head0;

findM(q,q0,aver);

sum=0;

printf("第1轮结束\n");

for(i=0;i<time-1;i++){

printf("第%d轮开始\n",(i+2));

//重新初始化数组

for(j=0;j<10;j++){

numArray[j]=0;

}

q=head;

while(q){

printf("%s请输入你的数:\n",q->name);

scanf\_s("%lf",&q->score);

while(isEqual(numArray,q->score)){

printf("已经被人输入，请重新输入：\n");

scanf\_s("%lf",&q->score,sizeof(q->score));

}

numArray[m]=q->score;

m++;

sum+=q->score;

q=q->next;

}

aver=sum/num\*0.618;

q=head;

q0=head0;

findM(q,q0,aver);

sum=0;

m=0;

printf("第%d轮结束\n",(i+2));

}

printf("游戏结束\n");

}

### 人机界面设计

（用图形展示该模块的用户界面。）

### 子程序设计



### 模块测试设计

（给出本模块的主要测试要求）

c++环境运行 支持window

## 模块2

-------

### 功能描述

将参与者输入的数字信息存入结构体数组中

### 接口描述

输入 每个参与者输入数字

输出 存入每个人的信息数组中

### 内部元素结构

（准确地描述每个模块包含的数据、子程序等。）

void goldPoint(){

int num;

int time;

double sum=0;

double aver;

int i,j;

int m=0;//用来初始化数组下标

scoreNode \*head,\*p,\*q,\*head0,\*q0,\*p0;

double numArray[10]={0};

head=NULL;

head0=NULL;

q=NULL;

q=head;

q0=head0;

printf("请输入参加的人数：\n");

scanf\_s("%d",&num);

printf("请输入需要进行几轮：\n");

scanf\_s("%d",&time);

printf("第1轮开始\n");

for(i=0;i<num;i++){

p=(scoreNode \*)malloc(sizeof(scoreNode));

p0=(scoreNode \*)malloc(sizeof(scoreNode));

printf("请输入你的名字：\n");

scanf\_s("%s",p->name,sizeof(p->name));

printf("请输入你的数：\n");

scanf\_s("%lf",&p->score,sizeof(p->score));<br>　　　　　　　　　　//每轮输入的数字不能相同

while(isEqual(numArray,p->score)){

printf("已经被人输入，请重新输入：\n");

scanf\_s("%lf",&p->score,sizeof(p->score));

}

numArray[i]=p->score;

p0->next=NULL;

p->next=NULL;

if(head==NULL){

head=p;

q=p;

}else{

q->next=p;

q=p;

}

sum+=p->score;

strcpy\_s(p0->name,strlen(p->name)+1,p->name);

p0->score=0;

if(head0==NULL){

head0=p0;

q0=p0;

}else{

q0->next=p0;

q0=p0;

}

}

aver=sum/num\*0.618;

//printf("aver=%lf\n",aver);

q=head;

q0=head0;

findM(q,q0,aver);

sum=0;

printf("第1轮结束\n");

for(i=0;i<time-1;i++){

printf("第%d轮开始\n",(i+2));

//重新初始化数组

for(j=0;j<10;j++){

numArray[j]=0;

}

q=head;

while(q){

printf("%s请输入你的数:\n",q->name);

scanf\_s("%lf",&q->score);

while(isEqual(numArray,q->score)){

printf("已经被人输入，请重新输入：\n");

scanf\_s("%lf",&q->score,sizeof(q->score));

}

numArray[m]=q->score;

m++;

sum+=q->score;

q=q->next;

}

aver=sum/num\*0.618;

q=head;

q0=head0;

findM(q,q0,aver);

sum=0;

m=0;

printf("第%d轮结束\n",(i+2));

}

printf("游戏结束\n");

}

### 人机界面设计

（用图形展示该模块的用户界面。）

### 子程序设计



### 模块测试设计

（给出本模块的主要测试要求）

c++环境运行 支持window

## 模块3

-------

### 功能描述

得出G值并且比较

### 接口描述

输入 没每个参与者输入数字

输出 得出G值并比较

### 内部元素结构

（准确地描述每个模块包含的数据、子程序等。）

int isEqual(double e[10],double temp){

//double \*p;

//p=e;

int i;

for(i=0;i<10;i++){

if(e[i]==temp)

return 1;

}

return 0;

}

### 人机界面设计

（用图形展示该模块的用户界面。）

### 子程序设计



### 模块测试设计

（给出本模块的主要测试要求）

c++环境运行 支持window

## 模块4

-------

### 功能描述

比较G值和输入值，增加或减少分数

### 接口描述

输入 没每个参与者输入数字

输出 增加或减少分数

### 内部元素结构

（准确地描述每个模块包含的数据、子程序等。）

void findM(scoreNode \*inLink,scoreNode \*scLink,double aver){

    char name[10];

    char name1[10];

    double temp0,temp1;

    scoreNode \*p=NULL;

    p=scLink;

    temp0=temp1=fabs(inLink->score-aver);

    strcpy\_s(name,strlen(inLink->name)+1,inLink->name);

    strcpy\_s(name,strlen(inLink->name)+1,inLink->name);

    inLink=inLink->next;

    //printf("%lf\n",inLink->score);

    while(inLink){

        double temp2=fabs(inLink->score-aver);

        if(temp0>temp2){

            temp0=temp2;

            strcpy\_s(name,strlen(inLink->name)+1,inLink->name);

        }

        if(temp1<temp2){

            temp1=temp2;

            strcpy\_s(name1,strlen(inLink->name)+1,inLink->name);

        }

        inLink=inLink->next;

    }

    while(scLink){

        if(strcmp(name,scLink->name)==0){

            scLink->score=scLink->score+1;

        }

        if(strcmp(name1,scLink->name)==0){

            scLink->score=scLink->score-1;

        }

        scLink=scLink->next;

    }

    while(p){

        printf("%s的得分为：%lf\n",p->name,p->score);

        p=p->next;

    }

}

### 人机界面设计

（用图形展示该模块的用户界面。）

### 子程序设计



### 模块测试设计

（给出本模块的主要测试要求）

c++环境运行 支持window