# 

# 软件工程专业2018级C语言课程设计

# 指

# 导

# 书

**软件学院 万琳**

**2018-11-1**

# 目 录

[寄 语 3](#_Toc345587130)

[C程序设计课程设计要求 4](#_Toc345587131)

[附件1: 6](#_Toc345587132)

[贪吃蛇游戏设计指导 6](#_Toc345587133)

[附件2： 12](#_Toc345587134)

[C语言编程风格规范 12](#_Toc345587135)

[附件3： 15](#_Toc345587136)

[课程设计报告书 15](#_Toc345587137)

# 寄 语

**各位同学，大家好！**

**C语言的学习也接近尾声，通过近一段时间的学习，我们了解了C语言的基本知识。但是对于我们大家来说，课程设计是一个新的开始！希望在4周的课程设计中，每位同学都能依靠自己的力量，我们共同努力，争取让我们的C语言的水平上一个台阶！**

**预祝大家取得好的成绩！**

**万琳**

**2018年11月11日**

# C程序设计课程设计要求

C语言课程设计环节旨在通过一个综合性实验让学生在充分掌握C语言程序理论知识的基础上，通过实践巩固所学知识提高编程能力和调试程序的能力。

**一、课程设计基本要求**

1、本次课程设计采取一人一组的方式，各人独立完成；

2、集成开发环境可选择Turbo C、Visual C++或者.Net框架等，但必须使用C语言进行程序的编写。

**二、课程设计进度安排**

1、第1周：

选题：列在本指导书“三、课程设计题目”中；

分析：在选题后对问题进行需求分析和定义，充分理解该选题的要求和内容。其基本任务是准确的回答“程序必须做什么？”这个问题。需求定义并不是确定程序如何完成它的工作，而是确定程序必须达到那些功能，也就是对目标程序提出完整、准确、清晰、具体的要求。

2、第2-3周：

设计：对算法和数据结构进行设计。这个阶段包括两方面内容：首先，在理解需求和分析需求的基础上，确定为了实现这些需求，目标程序所需要的数据结构，算法和程序流程。然后，对已经确定的数据结构，算法等进一步细化出所有的子功能，确定出模块间的接口，描述每个模块的处理过程，确定输入输出数据的内外部形式。

3、第4周：

实现：编程实现并反复调试。

提交：在指定时间（下学期会根据学生上课时间动态安排并提前通知）提交课程设计作业（内容参见本指导书“四、提交内容”），同时回答1-2个问题（将由包括教师在内的多名人员同时进行，每班由专人负责）。

**三、课程设计题目**

聪明的小蛇游戏（附件1）

这个选题主要针对游戏这一应用领域进行，完成该题目需要对链表、文件、键盘响应、图形编程等多方面的知识。希望同学们能在课程设计中学习到更多有用的知识！

**四、提交作业的时间要求和形式要求**

答辩时每位同学演示完成的系统并提交，同时提交课程设计的最终报告。

提交内容包括：

（1）电子版：每人提交一个文件夹，文件夹以学号加姓名命名，文件夹内包括源程序、可执行程序；

（2）纸质版：课程设计报告（要求手写，纸张必须选择16开信纸或大小相当的草稿纸）。

其中，源程序格式参见附件2，课程设计报告格式参见附件3。

**五、参考资料**

教材 《C程序设计》谭浩强编著。

# 附件1:

# 聪明的小蛇游戏设计指导

**一、简介**

有一条聪明的小蛇在草地上正在玩耍。草地上有美味的食物、有智慧草，小蛇玩得很开心。可是同时草地上也危机四伏，有地雷和会突然同时出现多个的毒草。可是小蛇非常聪明，每次它们都能逢凶化吉、遇难成祥。

1. **设计要求**
2. 在打开程序后，出现选择界面，里面包含难度（关卡）选择和退出功能，其中难度选择能够是程序出现相应的游戏难度，要求至少有三种。难度与以下因素相关：
3. 蛇的运动速度：难度越高，速度越快；
4. 地图不同：地图的边界可以不规则，难度越高，边界的变化越多；
5. 地雷和毒草的个数不同：难度越高，个数越多；
6. 开始游戏后，上下左右方向键控制小蛇的移动方向，小蛇的运动方式有两种选择。
7. 小蛇可以向上下左右移动，但是当小蛇向一个方向移动时，不能立刻向反方向移动。如小蛇正在向右移动，则必须向下或上移动一段后才能向左移动。
8. \*小蛇可以360度转动。
9. 小蛇撞到不同的东西会有不同的命运。
10. 美味的食物：吃到一个食物后，身体随之增加一个单位长度；
11. \*智慧草：智慧草每次出现一根，一段时间后消失。吃到智慧草后，小蛇无需控制可自动运动一段时间；
12. 地雷：小蛇踩到地雷会被炸掉一半，如果它当时身体的长度只有一节则会被炸死；
13. 毒草：毒草每次出现多根，\*保持闪烁状，一段时间后全部消失。小蛇吃到毒草后，身体随之减少一个单位长度。
14. 小蛇撞到墙时会死。
15. 当贪吃蛇身体长度达到一定时，游戏胜利可进入下一关卡，如果通过了最后一关则游戏胜利。
16. 需要对游戏情况进行计分，计分规则可自行拟定。
17. 提供排行榜功能，将游戏得分记载到排行榜文件中。
18. 提供进度保存功能，即中断游戏时可保存当前游戏状态，且要求将状态保存到文件中。至少要保存进入到哪一关，得分多少。\*另外可保存当时的游戏场景。
19. \*可以将地图保存在文件中，并且让用户选择地图开始游戏。
20. 界面美观，使用方便。\*最好用图形界面。

说明：以上标\*的部分为选做部分。

1. **所需数据结构分析**

**要求蛇必须采取数据结构1，功能键选择2中所示键，其余数据结构仅供参考。**

1. **蛇数据结构**

struct snake{/\*\*\*\*\*\*\*\*\*\*\*\*\*蛇结构体\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

char x;

char y;

struct snake \*previous;/\*前\*/

struct snake \*next; }\*head,\*tail;

1. **功能键的键值**

#define LEFT 0x4B00

#define RIGHT 0x4D00

#define UP 0x4800

#define DOWN 0x5000

#define ESC 0x011B

#define ENTER 0x1C0D

1. 游戏取界面数据

Char map[30][30] /\*游戏界面为30\*30\*/

#define STARTX 50 /\*表示游戏区在屏幕上的起始X坐标\*/

#define STARTY50 /\*表示游戏区在屏幕上的起始Y坐标\*/

#define SIZEX 20 /\*表示一个方块的长度\*/

#define SIZEY 20 /\*表示一个方块的宽度\*/

1. 游戏状态数据

Int gameRes /\*记录游戏的结果状态，值为0表示按ESC键退出；-1表示游戏失败；1表示游戏胜利\*/

1. **游戏流程图**

开始

图形方式初始化

初始化新游戏

从键盘读入操作信息

各类消息判断并作相应的处理

重新游戏?

结束

是

否

对“各类消息判断并作相应的处理”步骤的流程图细化：

Switch(key){

case ENTER:

case ESC:

gameRes=0;

case LEFT:

moveleft(); break;

case RIGHT:

moveright(); break;

case UP:

moveup(); break;

case DOWN:

movedown; break;

default: beeak;

}

1. **所需算法分析**
   1. 写出主程序的基本框架，确定函数各个内部模块的函数名和参数

int main()

{

initGraph(); /\*图形显示方式初始化\*/

{

Newgame(); /\*开局\*/

Int gameRes=0;

do

{

int key=getkey(); /\*读入操作信息\*/

if(key==ESC)

{

gameRes=0;break;

}

Switch(key)

{

对其他key值作相应的处理

}

If （checkwin()） /\*判断游戏是否胜利\*/

gameRes=1;

}while(!gameRes);

}while(!confirm(gameRes));

Return 0;

}

* 1. 图形初始化

void s\_detectgraph() {/\*自定义图形检测\*/

int gdriver,gmode,errorcode;

gdriver=VGA;

gmode=VGAHI;

registerbgidriver(EGAVGA\_driver);

initgraph(&gdriver,&gmode,"");

errorcode = graphresult();

if (errorcode !=0){

printf("\n\t\t\tGame : Snake Game\n");

printf("\n\terror:\t%s\n", grapherrormsg(errorcode));

getch();

exit(1);

}

}

* 1. 读入操作信息的处理

Switch(key){

case ENTER:

case ESC:

gameRes=0;

case LEFT:

moveleft(); break;

case RIGHT:

moveright(); break;

case UP:

moveup(); break;

case DOWN:

movedown; break;

default: beeak;

1. **关键函数介绍**
   1. 蛇链表

用链表的节点表示蛇，移动的时候增加一个头节点去掉一个尾节点，吃食物只需增加一个头节点。

* 1. barbox函数

void barbox(int x,int y,int color,int width,int height){/\*填充一定范围的函数\*/

setfillstyle(SOLID\_FILL,color); bar(x,y,x+width-1,y+height-1);

}

* 1. bioskey()

功 能: 直接使用BIOS服务的键盘接口

　　 函数原型：int bioskey (int cmd)

　　 说明：bioskey()的函数原型在bios.h中

　　 bioskey()完成直接键盘操作，cmd的值决定执行什么操作。

　　 cmd = 0：

　　 当cmd是0，bioskey()返回下一个在键盘键入的值（它将等待到按下一个键）。它返回一个16位的二进制数，包括两个不同的值。当按下一个普通键时，它的低8位数存放该字符的ASCII码；对于特殊键（如方向键、F1～F12等等），低8位为0，高8位字节存放该键的扫描码。

　　 cmd = 1：

　　 当cmd是1，bioskey()查询是否按下一个键，若按下一个键则返回非零值，否则返回0。

* 1. setviewport()

功 能: 为图形输出设置当前视口

　 用 法: void far setviewport(int left, int top, int right, int bottom, int clipflag);

5、settextstyle()

功 能: 为图形输出设置当前的文本属性

　　 用 法: void far settextstyle (int font, int direction, char size);

　　fornt 为字体：DEFAULT\_FONT， TRIPLEX\_FONT， SMALL\_FONT， SANSSERIF\_FONT， GOTHIC\_FONT，也可以用0 ~ 4代替。

　 　direction 为字符的排列方向：横向和竖向，0为横向排列，1为竖向排列。

　 　size 为字体大小：可用interger做参数。

6、outtextxy（）

　　 功 能: 在指定位置显示一字符串

　　 用 法: void far outtextxy(int x, int y, char \*textstring);

# 附件4：

# C语言编程风格规范

良好的程序编写风格将有助于提高程序编写与调试的效率，改善程序的可读性，增强程序的可维护性。

1. 大括号的位置

将开始的大括号放在一行的最后，将结束大括号放在一行的第一位。如下所示：

If(x is ture){

We are do y

}

命名函数时：

Int function(x){/\*函数的功能\*/

Body of function；

}

1. 缩进格式

一行只写一个语句并使用Tab进行缩进，Tab是8个字符的缩进。

1. 命名系统

定义的变量名等标识符名称应尽量接近其用途或意义。比如逆序函数int reverse（）；当需要混合类型的名字时候，那么需要使用描述性的命名方式。如有一个函数用来计算活动用户的数量应该这样命名“count\_active\_users()”.

1. 函数

函数应该短小且目的明确，一个函数只做一件事情。

1. 注释

关键的标识符、语句、函数和程序段应给出注释，说明其功用。如：

void box\_mark(int x,int y){/\*给小方块插旗标记\*/

int nx=ha\_x+2+x\*10,ny=ha\_y+2+y\*10;

if(map[y][x]>>1&1)return;/\*已经打开\*/

if(map[y][x]>>2&1)map[y][x]-=1<<2;/\*有旗拔旗\*/

else map[y][x]+=1<<2;/\*插旗\*/

。。。

}

实例：输入某年某月某日，判断这一天是这一年的第几天？

/\*输入某年某月某日，判断这一天是这一年的第几天？\*/

main()  
{  
int day,month,year,sum,leap;  
printf("\nplease input year,month,day\n");  
scanf("%d,%d,%d",&year,&month,&day);  
switch(month) {/\*先计算某月以前月份的总天数\*/  
　case 1:sum=0;break;  
　case 2:sum=31;break;  
　case 3:sum=59;break;  
　case 4:sum=90;break;  
　case 5:sum=120;break;  
　case 6:sum=151;break;  
　case 7:sum=181;break;  
　case 8:sum=212;break;  
　case 9:sum=243;break;  
　case 10:sum=273;break;  
　case 11:sum=304;break;  
　case 12:sum=334;break;  
　default:printf("data error");break;  
 }  
sum=sum+day; 　/\*再加上某天的天数\*/  
　if(year%400==0||(year%4==0&&year%100!=0))/\*判断是不是闰年\*/  
　　leap=1;  
　else  
　　leap=0;  
if(leap==1&&month>2)/\*如果是闰年且月份大于2,总天数应该加一天\*/  
sum++;  
printf("It is the %dth day.",sum);

}

# 附件3：

# 课程设计报告书

* 1. 选题

给出选题名称即可。

二、需求分析

对问题进行需求分析和定义，明确回答“程序必须做什么？”这个问题，确定程序必须达到那些功能，对目标程序提出完整、准确、清晰、具体的要求。

三、系统设计

在理解需求和分析需求的基础上，确定为了实现这些需求，目标程序所需要的数据结构，算法和程序流程。然后，对已经确定的数据结构，算法等进一步细化出所有的子功能，确定出模块间的接口，描述每个模块的处理过程，确定输入输出数据的内外部形式。

1、数据结构设计：列出关键的数据结构；

2、算法和程序流程：画出相应程序流程图；

3、模块间接口描述。

四、界面设计

对界面或交互方式进行设计并给出相应图解。

五、系统实现

列出编程草稿（至少包括核心程序），可用源代码或伪代码列出。

六、测试与调试

记录测试，调试和程序修改的过程。

七、心得与体会