吉林大学

2016-2017学年第二学期

《C语言程序设计》

项目文档

第12组

|  |  |  |
| --- | --- | --- |
| 项目题目 | 网格化小游戏组合 | |
| 组长 | | |
| 王文楷 | 33160721 | 化学 |
| 组员 | | |
| 张润泽 | 33160620 | 化学 |
| 周元尊 | 34160705 | 生物 |
| 李冰文 | 33160520 | 化学 |
| 1. 总体设计   考虑到平时接触到的小游戏比较多，且可以学习到的代码较多，并且由于接触的比较多，对这类游戏中的规则了解的比较多，对程序的整体思路有着相对较好的把握，能够保证我们在进行对游戏规则的编写上以及讨论中能够集思广益。  设计思路：  总体：利用gotoxy函数控制输出位置，kbhit函数判断键盘输入状态  “消消乐”：通过建立两个二维数组，一个用来存放格子中的颜色，一个用来存放该格子的状态，从而判断能否完成消除。  Flappy-Bird：通过判断小鸟是否与柱子重合进行游戏。  扫雷：建立三个二维数组，一个存放是否有雷，一个存放周围雷数，另一个存放该格子是否被揭示。  贪吃蛇：建立一个二维数组储存地图信息（蛇，空格，障碍），一个一位数组储存蛇的历史位置。  五子棋：建立一个二维数组储存棋盘信息。  界面：实现点击图标跳转至游戏。 | | | |
| 1. 主要功能和亮点   ①扫雷：  可实现键盘控制扫雷，规则与原版扫雷基本一致。  创新设计了高亮单元格以替代鼠标指针，以及利用键盘控制上下左右（WSAD）移动改变位置，既避免了鼠标操作程序语言的复杂，又保证了玩家操作便捷。使用了gotoxy函数调整光标位置，从而不用每次清屏浪费时间且造成屏幕闪烁。  改变主要为键盘操作模式。在一局游戏结束后玩家可通过按y或n选择重新开始或退出，从而避免程序反复关闭再运行浪费时间。画面上仍需优化。原游戏中鼠标右键多次点击可以做到标记，标疑，取消功能，由于较复杂在本程序中取消。  ②贪吃蛇：  可实现键盘控制贪吃蛇，规则与原版贪吃蛇基本一致（不可穿墙）。  创新设计了自定义地图，通过txt文档导入，只需修改程序中的一个字符即可体验多样的自定义地图，使游戏不至于太单一。玩家可通过按空格键暂停游戏，并再按一次恢复。另外，记录蛇历史位置的数组采用环形的重复利用方式（即存储至最后一个单元后跳转回第一个单元），从而避免空间和时间的浪费。  在一局游戏结束后玩家可通过按y或n选择重新开始或退出，从而避免程序反复关闭再运行浪费时间。  ③消消乐（3消）  可实现键盘控制消除，只包含基础功能（没有奖励道具，地形障碍和计分系统）。  创新和改变方面基本同扫雷。  ④五子棋（pvp模式）  可实现键盘控制双人五子棋，规则与原版一致。  创新和改变方面基本同扫雷。另外做了界面的优化，设置了背景颜色，画面大小，优化了游戏体验。  ⑤flappy bird  可实现键盘控制小鸟移动，规则与原版一致。  功能上暂无创新和改动。  主要改变了现在C语言通用的清屏函数而通过自己创建点并在下一个循环中擦除  有效的简化了原函数的计算。  ⑥界面，由于希望开发一个较好的界面,自学了MFC部分功能去改观现有界面。 | | | |
| 1. 组员分工   王文楷：游戏程序的主要负责人，为他人的程序提供改进以及在难度较大的函数上提供协助。  李冰文：Flappy-Bird和界面部分的负责人。  张润泽：“消消乐”部分的负责人。  周元尊：“五子棋”部分的负责人。 | | | |
| 1. 程序代码使用说明   所有程序只需要c语言编译程序即可运行。  代码库主要包括： stdio.h stdlib.h windows.h conio.h。含有以下内容：时间指针、初始化随机种子函数、光标和字体颜色控制函数、检测键盘输入函数等。(MFC中另算) | | | |
| 1. 主要代码   （亮点部分的带有注释的实现代码，不要太多；填写时删除此行）  ①扫雷中的键盘控制的程序段，其他程序中键盘控制与之类似不再展示。  do  {  if(kbhit())  {  f1=0;  input=getch();  switch(input)  {  case 'w':clear(x,y); if(x>1) x--; highlight(x,y); break;//move the location of the blue frame that marks the tile you’re operating on  case 's':clear(x,y); if(x<16) x++; highlight(x,y); break;  case 'a':clear(x,y); if(y>1) y--; highlight(x,y); break;  case 'd':clear(x,y); if(y<30) y++; highlight(x,y); break;  case 'j':f1=1; break;//pressing 'j' is the same as clicking the left side of the mouse(you think a tile as safe)  case 'k':f1=1; break;//pressing 'k' is the same as clicking the right side of the mouse(you think there's a mine in the tile)  case ' ':f1=1; break;//pressing ' ' is the same as clicking both sides of the mouse(you think the mines around the tiles have all been detected) this kind of move doesn't kill you  }  }  }while (f1==0||input!=' '&&c[x][y]==1||input==' '&&c[x][y]==0); //some moves are invalid  ②扫雷中判断正确时揭示格中内容的函数，a数组存放某格中是否有雷（0无1有），b数组存放周围雷数，本身是雷则储存-1，c数组存放某格是否被揭示（0否1是）。该函数通过递归可以实现一片区域的揭示。  void reveal(int i,int j,char input,int a[18][32],int b[18][32],int c[18][32])// reveal a tile when the player judged it right  {  if(input!=' ')  {  c[i][j]=1;  textcolor(10);  gotoxy(5\*j,2\*i);  if(a[i][j]==1) printf("%2c",6); else printf("%2d",b[i][j]);  if(input=='j'||input==' ') delay(30); else delay(200);  switch(b[i][j])  {  case 0: textcolor(8); break;  case -1:textcolor(15); break;  case 1: textcolor(11); break;  case 2: textcolor(10); break;  case 3: textcolor(12); break;  case 4: textcolor(9); break;  case 5: textcolor(13); break;  case 6: textcolor(14); break;  case 7: textcolor(5); break;  case 8: textcolor(6); break;  }  gotoxy(5\*j,2\*i);  if(a[i][j]==1) printf("%2c",6); else printf("%2d",b[i][j]);  }  if(input=='j'&&b[i][j]==0||input==' ')//if a number '0' is revealed or the player correctly pressed ' ', the tiles around the tile will be revealed  {  if(c[i-1][j]==0) reveal(i-1,j,'j',a,b,c);  if(c[i-1][j-1]==0) reveal(i-1,j-1,'j',a,b,c);  if(c[i-1][j+1]==0) reveal(i-1,j+1,'j',a,b,c);  if(c[i+1][j]==0) reveal(i+1,j,'j',a,b,c);  if(c[i+1][j-1]==0) reveal(i+1,j-1,'j',a,b,c);  if(c[i+1][j+1]==0) reveal(i+1,j+1,'j',a,b,c);  if(c[i][j-1]==0) reveal(i,j-1,'j',a,b,c);  if(c[i][j+1]==0) reveal(i,j+1,'j',a,b,c);  }  }  ③贪吃蛇中蛇身移动的片段（当移动后什么也不发生时）。a数组储存地图信息（空白为0，有蛇为1，有食物为2，障碍为3），b数组是创新点，储存蛇历史位置（百位以上是横坐标，以下是纵坐标），采用环形的重复利用方式（即存储至最后一个单元b[599]后跳转回第一个单元b[0]），从而避免空间和时间的浪费。b[p1]，b[p2]分别是b数组实际储存部分开始和结束（蛇尾和蛇头）的单元，i,j是当前蛇头位置  textcolor(7);  gotoxy(5\*(b[\*p1]%100),2\*(b[\*p1]/100));  printf(" ");  a[b[((\*p1)+1)%600]/100][b[((\*p1)+1)%600]%100]=0;  gotoxy(5\*(\*j),2\*(\*i));  printf("%2c",6);  gotoxy(5\*(b[\*p2]%100),2\*(b[\*p2]/100));  printf("%2c",1);  \*p1=((\*p1)+1)%600;\*p2=((\*p2)+1)%600;  b[\*p2]=100\*(\*i)+(\*j);  a[\*i][\*j]=1;  ④消消乐中消除和掉落部分，通过函数间的循环调用实现功能。  char scan(char a[18][32],char b[18][32]) //check all tiles that need examination and return a sign of whether any tiles can be removed{  char f;  int i,j;  for(i=1;i<=16;i++)  for(j=1;j<=30;j++)  if(b[i][j]==1) f+=check2(a,b,i,j);  if(f) eliminate(a,b);  return f;  }  ………  void eliminate(char a[18][32],char b[18][32]) //move away the marked tiles  {  int i,j,k;  delay(t2);  for(i=1;i<=16;i++)  for(j=1;j<=30;j++)  if(b[i][j]==2)  {  highlight(i,j,'g'); //the eliminated tiles are marked with green frame  a[i][j]=0;  gotoxy(4\*j+2,2\*i);  printf(" ");  }  delay(t2);  for(i=1;i<=16;i++)  for(j=1;j<=30;j++)  if(b[i][j]==2) highlight(i,j,'w'); //change the frame back to white  delay(t2);  for(i=16;i>=1;i--)  for(j=1;j<=30;j++)  if(b[i][j]==2)  {  for(k=i;k>=1;k--) b[k][j]=1; //all the upper tiles of the eliminated tile needs to be checked next time, marked with 1  }  fall(a,b); //the tiles fall  scan(a,b); //the fallen tiles might make other matches  }  ⑤flappy bird——这是函数中比较核心的部分，用于画出柱子并擦除。采用了比较节约运算的方式。  int barshow(int q,int i)  {  Sleep(1000);  int a,f;  if(q>=0)  {  f=i;  for(;f>=1;f--)  {  gotoxy(q+1,f);  printf(" ");  }  a=i+6;  for(;a<=high;a++)  {  gotoxy(q+1,a);  printf(" ");  }  f=i;  for(;f>=1;f--)  {  gotoxy(q,f);  printf("\*");  }  a=i+6;  for(;a<=high;a++)  {  gotoxy(q,a);  printf("\*");  }  f=i;  for(;f>=1;f--)  {  gotoxy(0,f);  printf(" ");  }  a=i+6;  for(;a<=high;a++)  {  gotoxy(0,a);  printf(" ");  }  if(bird\_y==(q+1))  {  if((bird\_x>i)&&(bird\_x<(i+6)))  score++;  else  {  printf("defeat\n");  system("pause");  exit(0);  }  }  gotoxy(0,30);  printf("score:%d",score);  }  return q;  }  ⑥五子棋中键盘控制和判断部分  void control(int\*count, int p[Q][Q], int \*Cx, int \*Cy, int \*Now)//键盘控制及判定结果  {  char input='1';  int victor;  int f = 1;  while (f)//循环变量，如果一局游戏判定结束，则不会继续判定  {  if (\_kbhit())  {  input = \_getch();//利用input将键盘所对应ASCII码输入，以实现键盘控制  if (input == 27)  exit(0);  if (input == 32)  {  if (changechess(p, \*Cx, \* Cy, \* Now))  {  (\*count)++;//记录棋子数    if (\*count == Q\*Q)//平局判定  {  gotoxy( Q\*2, Q);  printf("平局");  if (\_getch() == 224)//当一局游戏结束后，按任意键跳出  {  \_getch();  }    }  victor = Rule(p,\* Cx,\* Cy, \*Now);  if (victor == 1)//黑棋胜判定  {  gotoxy(Q\*2, Q);  printf("黑棋胜");  if (\_getch() == 224)  {  \_getch();  }    }  if (victor == 2)//白棋胜判定  {  gotoxy(Q\*2, Q);  printf("白棋胜");  if (\_getch() == 224)  {  \_getch();  }    }  \*Now = 3 - \*Now;  if (\*count == Q\*Q || victor == 1 || victor == 2)//改变循环变量，跳出循环  {  f = 0;    }  }  }  ⑦这个用于新建控件和按钮窗体。  void CMFCApplication6Dlg::DoDataExchange(CDataExchange\* pDX)  {  CDialogEx::DoDataExchange(pDX);  DDX\_Control(pDX, IDC\_BUTTON2, m2);  DDX\_Control(pDX, IDC\_BUTTON3, m3);  DDX\_Control(pDX, IDC\_BUTTON4, m4);  DDX\_Control(pDX, IDC\_BUTTON5, m5);  DDX\_Control(pDX, IDC\_BUTTON6, m6);  DDX\_Control(pDX, IDC\_BUTTON7, m7);  DDX\_Control(pDX, IDC\_BUTTON8, m8);  }  BEGIN\_MESSAGE\_MAP(CMFCApplication6Dlg, CDialogEx)  ON\_WM\_SYSCOMMAND()  ON\_WM\_PAINT()  ON\_WM\_QUERYDRAGICON()  ON\_BN\_CLICKED(IDC\_BUTTON2, &CMFCApplication6Dlg::OnBnClickedButton2)  ON\_BN\_CLICKED(IDC\_BUTTON3, &CMFCApplication6Dlg::OnBnClickedButton3)  ON\_BN\_CLICKED(IDC\_BUTTON4, &CMFCApplication6Dlg::OnBnClickedButton4)  ON\_BN\_CLICKED(IDC\_BUTTON8, &CMFCApplication6Dlg::OnBnClickedButton8)  ON\_BN\_CLICKED(IDC\_BUTTON5, &CMFCApplication6Dlg::OnBnClickedButton5)  ON\_BN\_CLICKED(IDC\_BUTTON7, &CMFCApplication6Dlg::OnBnClickedButton7)  END\_MESSAGE\_MAP()  用于改变按钮形态  BOOL CMFCApplication6Dlg::OnInitDialog()  {  CDialogEx::OnInitDialog();  // 将“关于...”菜单项添加到系统菜单中。  // IDM\_ABOUTBOX 必须在系统命令范围内。  ASSERT((IDM\_ABOUTBOX & 0xFFF0) == IDM\_ABOUTBOX);  ASSERT(IDM\_ABOUTBOX < 0xF000);  CMenu\* pSysMenu = GetSystemMenu(FALSE);  if (pSysMenu != NULL)  {  BOOL bNameValid;  CString strAboutMenu;  bNameValid = strAboutMenu.LoadString(IDS\_ABOUTBOX);  ASSERT(bNameValid);  if (!strAboutMenu.IsEmpty())  {  pSysMenu->AppendMenu(MF\_SEPARATOR);  pSysMenu->AppendMenu(MF\_STRING, IDM\_ABOUTBOX, strAboutMenu);  }  }  // 设置此对话框的图标。 当应用程序主窗口不是对话框时，框架将自动  // 执行此操作  SetIcon(m\_hIcon, TRUE); // 设置大图标  SetIcon(m\_hIcon, FALSE); // 设置小图标  // TODO: 在此添加额外的初始化代码  HBITMAP hBmp = ::LoadBitmap(AfxGetInstanceHandle(), MAKEINTRESOURCE(IDB\_BITMAP3));  m2.SetBitmap(hBmp);  HBITMAP hBm2 = ::LoadBitmap(AfxGetInstanceHandle(), MAKEINTRESOURCE(IDB\_BITMAP2));  m3.SetBitmap(hBm2);  HBITMAP hBm3 = ::LoadBitmap(AfxGetInstanceHandle(), MAKEINTRESOURCE(IDB\_BITMAP4));  m4.SetBitmap(hBm3);  HBITMAP hBm5 = ::LoadBitmap(AfxGetInstanceHandle(), MAKEINTRESOURCE(IDB\_BITMAP5));  m7.SetBitmap(hBm5);  HBITMAP hBm6 = ::LoadBitmap(AfxGetInstanceHandle(), MAKEINTRESOURCE(IDB\_BITMAP6));  m8.SetBitmap(hBm6);  return TRUE; // 除非将焦点设置到控件，否则返回 TRUE  }  用于添加事件。  HCURSOR CMFCApplication6Dlg::OnQueryDragIcon()  {  return static\_cast<HCURSOR>(m\_hIcon);  }  void CMFCApplication6Dlg::OnBnClickedButton2()  {  // TODO: 在此添加控件通知处理程序代码  WinExec("fly.exe", SW\_SHOW);  }  void CMFCApplication6Dlg::OnBnClickedButton3()  {  // TODO: 在此添加控件通知处理程序代码  WinExec("wwk1.exe", SW\_SHOW);  }  void CMFCApplication6Dlg::OnBnClickedButton4()  {  WinExec("zyz.exe", SW\_SHOW);  }  void CMFCApplication6Dlg::OnBnClickedButton8()  {  // TODO: 在此添加控件通知处理程序代码  WinExec("zrz.exe", SW\_SHOW);  }  void CMFCApplication6Dlg::OnBnClickedButton5()  {  // TODO: 在此添加控件通知处理程序代码  WinExec("zrz.exe", SW\_SHOW);  }  void CMFCApplication6Dlg::OnBnClickedButton7()  {  // TODO: 在此添加控件通知处理程序代码  WinExec("snake.exe", SW\_SHOW);  } | | | |
| 1. 程序运行截图   ①扫雷      ②贪吃蛇      ③消消乐    ④五子棋    ⑤flappy bird    ⑥界面 | | | |
| 1. 收获与感悟   在学业方面：通过课堂上的学习，基础的把握了函数的设计，以及对程序设计的主体思路的把握有了比较明晰的认识。学会了debug的能力。  通过多加练习才能提高熟练度而不是仅仅在课堂上掌握了理论知识就能把一个程序成功编写出来，经常会在思路明晰的情况下发现自己的函数编写出了问题，其实是对函数结构的不了解以及函数本身的用途把握不够准确。  通过在project的构建上，学会了在网上挑选合适自己的函数，以及比较熟练的自学能力。通过编写一个程序，提高了自己在处理问题上的一个整体性的把握和全局观的培养。  这也算是一个向他人请教的过程，通过向本专业的人学习能够得到网上并不能讲的全面的小知识。  通过团队的合作，能够真正的了解到团队合作以及团队分工的意义，在向他人请教的过程中能够提高自己对一个程序的理解。 | | | |
| 1. 意见与建议   与国际化课程接轨十分新颖并且十分有用，能够让学生对C语言有着更正规和更国际化的认识，并为日后更深入的学习打下了一定的基础。  内容相对全面,对基础的教学比较完整，方便了学生能够继续自学。  然而教学进度相对较快，基础薄弱的学生比较辛苦，希望能够在上学期就能够普及或者布置下一些基础知识教学的任务，以方便学生了解。 | | | |