**实验报告**

年 月 日 成绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓名 | 刘济铭 | 学号 | 16201121 | 班级 | 16052311 |
| 专业 | 计算机科学与技术 | | 课程名称 | 嵌入式系统课程设计 | |
| 任课老师 | 严义 | 指导老师 | 严义 | 机位号 |  |
| 实验序号 | 6 | 实验名称 | Keil环境的装载和了解 | | |
| 实验时间 |  | 实验地点 | 1教604 | 实验设备号 |  |
| **一、实验目的和实验要求**  1.设计实现对子棋的操作步骤  2.加入简单的人工智能算法（智障搜索算法）   1. **实验步骤**   1.根据对弈规则，实现棋子与棋盘显示  2.实现双方对弈的过程  3.加入简单的搜索算法，实现一方棋子的自动走位   1. **实验结果**     int main(void)  {  TOUCH\_SCREEN\_INIT();  LCD\_Init();  tim\_init();    TOUCH\_INT\_config();  TOUCH\_INT\_EXIT\_Init();  TOUCH\_InterruptConfig();    Draw\_chessboard();  while (1)  {  }  }    //对弈程序  static void LCD\_BIG\_POINT(u16 x, u16 y)  {  int i = (int)(x/40)-1;  int j = (int)(y/40)-1;  int flag=0;  int number = 0;  int iii=0,jjj=0,index=0;  int Score[20][5];  int flag1=0;    //ÖØÐÂÑ¡×Ó  if(Chessman[i][j] == Chessman[I][J] &&SW!=0 && sum>0)  {  if(Chessman[I][J]==-1){  LCD\_Draw\_Circle((I+1)\*40,(J+1)\*40,10,1,BLACK);  LCD\_Color\_Fill(219,299,239,319,BLACK);  LCD\_Color\_Fill(0,0,20,20,YELLOW);  LCD\_Draw\_Circle(x,y,10,1,RED);  Chessman[I][J]=-1;  I=i;J=j;}  else{  LCD\_Draw\_Circle((I+1)\*40,(J+1)\*40,10,1,WHITE);  LCD\_Color\_Fill(0,0,20,20,WHITE);  LCD\_Color\_Fill(219,299,239,319,YELLOW);  LCD\_Draw\_Circle(x,y,10,1,RED);  Chessman[I][J]=1;  I=i;J=j;}  }    //Âä×Ó  else if(Chessman[i][j]==0 && SW != 0)  {  if(abs(i-I)+abs(j-J)==1)  {  // if(SW==-1 && sw!=-1)  // {  // Run\_astep(x,y,i,j,-1);  // }    if(SW==1 && sw!=1)  {  Run\_astep(I,J,i,j,1);    for(ii=0;ii<5;ii++)  {  for(jj=0;jj<7;jj++)  {  if(Chessman[ii][jj]==-1)  {  if(ii<4){  if(Chessman[ii+1][jj]==0)  {  Chessman[ii+1][jj]=-1;  Chessman[ii][jj]=0;  flag = Killtest(ii+1,jj);  Chessman[ii+1][jj]=0;  Chessman[ii][jj]=-1;    Score[index][0] = ii;Score[index][1] = jj;Score[index][2] = ii+1;Score[index][3] = jj;Score[index][4] = flag;  index++;  }  }  if(jj<5){  if(Chessman[ii][jj+1]==0)  {  Chessman[ii][jj+1]=-1;  Chessman[ii][jj]=0;  flag = Killtest(ii,jj+1);  Chessman[ii][jj+1]=0;  Chessman[ii][jj]=-1;    Score[index][0] = ii;Score[index][1] = jj;Score[index][2] = ii;Score[index][3] = jj+1;Score[index][4] = flag;  index++;  }  }  if(ii>0){  if(Chessman[ii-1][jj]==0)  {  Chessman[ii-1][jj]=-1;  Chessman[ii][jj]=0;  flag = Killtest(ii-1,jj);  Chessman[ii-1][jj]=0;  Chessman[ii][jj]=-1;    Score[index][0] = ii;Score[index][1] = jj;Score[index][2] = ii-1;Score[index][3] = jj;Score[index][4] = flag;  index++;  }  }    if(jj>0){  if(Chessman[ii][jj-1]==0)  {  Chessman[ii][jj-1]=-1;  Chessman[ii][jj]=0;  flag = Killtest(ii,jj-1);  Chessman[ii][jj-1]=0;  Chessman[ii][jj]=-1;    Score[index][0] = ii;Score[index][1] = jj;Score[index][2] = ii;Score[index][3] = jj-1;Score[index][4] = flag;  index++;  }  }  }}}  index--;  flag=-1;  for(ii=0;ii<=index;ii++)  {  if(Score[ii][4]==1) {I =Score[ii][0];J=Score[ii][1];iii = Score[ii][2];jjj=Score[ii][3];flag=1;flag1=1;}  if(Score[ii][4]==0 && flag1<1) flag=0;  }  if(flag == 0)  {  ii = 0;  flag1=0;  for(ii=0;ii<=index;ii++)  {  if(Score[ii][4]==0) flag1++;  }  ii = 0;  number = count%flag1;  while(number>-1)  {  if(Score[ii][4]==0) number-=1;  ii++;  }  I =Score[ii][0];  J=Score[ii][1];  iii = Score[ii][2];  jjj=Score[ii][3];  }  if(flag == -1)  {  ii = 0;  number = 2\*(count%5);  while(number>-1)  {  if(Score[ii][4]==-1) number-=1;  ii++;  }  I =Score[ii][0];  J=Score[ii][1];  iii = Score[ii][2];  jjj=Score[ii][3];  }  flag1=0;flag=0;index=0;  Run\_astep(I,J,iii,jjj,-1);  SW=0;sw=-1;  }  }  }    //Ñ¡×Ó  else if(Chessman[i][j]==1 && SW ==0)  {  if(y>=40 && x<=280 && sum==0)  {  SW = Chessman[i][j]; //Âä×ÓÑÕÉ«  LCD\_Draw\_Circle(x,y,10,1,RED);  if(SW==1) LCD\_Color\_Fill(0,0,20,20,WHITE);  else LCD\_Color\_Fill(219,299,239,319,BLACK);  I=i;J=j;  }  else if(y>=40 && x<=280&&sum>0)  {  SW = Chessman[i][j];  LCD\_Draw\_Circle(x,y,10,1,RED);  I=i;J=j;  }  }  } u32 xScreen, yScreen; | | | | | |