附件1：实验报告中的诚信设计

数据结构与算法导论

实验报告

实验题目：\_\_\_\_\_人机对战迷宫游戏\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

姓 名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学 号：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

日 期：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

自我评分：\_\_\_\_\_\_\_\_\_【 X 】\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

自我评分说明：A+，A，B+，B，B-，C，D，分别对应分数95、90、85、80、75、70、60

诚信声明

本人郑重承诺：本实验程序和实验报告均是本人独立学习和工作所获得的成果。尽我所知，实验报告中除特别标注的地方外，不包含其他同学已经发表或撰写过的成果；实验程序中对代码工作的任何帮助者所作的贡献均做了明确的说明，并表达了谢意。

如有抄袭，本人原因承担因此而造成的任何后果。

特此声明。

签名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

日期：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

程序引用说明

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 引用项 | 来源 | 代码引用行数 |
| 1 | 控制函数 | CSDN | 50 |
| 2 | 迷宫数组 | CSDN | 20 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 小计 | | | 70 |

总代码行数\_\_\_\_\_\_\_\_\_\_402\_\_\_\_\_\_\_\_\_\_; 引用占比\_\_\_\_\_\_17%\_\_\_\_\_\_\_\_\_\_

1、实验简介

受给定题目二俄罗斯方块人机对战启发，根据这学期对于图的学习思考，有了做一个走迷宫人机对战的想法。

程序在菜单处有两个选项，分别可以进行自己绘制迷宫，让机器作答并给出路径；或者机器给出迷宫，人来作答。

2、程序框架

//2种迷宫画图函数

void draw1(void);

void draw2(void);

//控制函数

void control1(void);

void control2(void);

//菜单函数

void menu(void);

//移动后位置交换函数

void move(int \*i, int \*j);

int bfs();//进行机器走迷宫运算

int judge(int a, int b);//判断所在点是否出边界，出界返回0，未出界返回1

3、关键代码实现

【哪些函数是你认为最能体现自己工作成果的函数，说明函数实现基本思想（可用文字或图表示），以及具体的实验步骤（用伪代码或带注释代码）】

3.1 bfs()

int bfs()

{

pos cur, nex;

cur.a = fa;

cur.b = fb;

cur.steps = 0;

memset(cur.Mapp, 0, sizeof(cur.Mapp));

cur.Mapp[cur.a][cur.b] = 1; //初始化起点坐标

queue<pos>qu;

qu.push(cur);

vis[cur.a][cur.b] = 1;

while (!qu.empty())

{

cur = qu.front();

qu.pop();

if (cur.a == aa && cur.b == ab)

{

memcpy(Mappp, cur.Mapp, sizeof(cur.Mapp));

return cur.steps;

}

for (int i = 0; i < 4; i++)

{

nex.a = cur.a + dir[i][0];

nex.b = cur.b + dir[i][1];

nex.steps = cur.steps + 1;

memcpy(nex.Mapp, cur.Mapp, sizeof(cur.Mapp));

nex.Mapp[nex.a][nex.b] = 1;

if (vis[nex.a][nex.b] == 0 && judge(nex.a, nex.b))

{

vis[nex.a][nex.b] = 1;

qu.push(nex);

}

}

}

return inf;

}

此函数通过bfs搜索找到通往出口的最短路径并且进行存储

3.2 void control2(void)

void control2(void)

{

int a, b;

char num;

do { //每一次移动前找到小人的位置

for (int i = 0; i < 20; i++)

for (int j = 0; j < 20; j++)

if (s2[i][j] == 2)

{

a = i;

b = j;

break;

}

//接收控制

num = getch();

switch (num)

{

case 72:if (s2[a - 1][b] != 1) //向上

{

move(&s2[a][b], &s2[a - 1][b]);

system("cls");

draw2();

break;

}

else continue;

case 80:if (s2[a + 1][b] != 1)//向下

{

move(&s2[a][b], &s2[a + 1][b]);

system("cls");

draw2();

break;

}

else continue;

case 75:if (s2[a][b - 1] != 1) //向左

{

move(&s2[a][b], &s2[a][b - 1]);

system("cls");

draw2();

break;

}

else continue;

case 77:if (b == 19) continue;//向右，小人不能走出地图

else if (s2[a][b + 1] != 1)

{

move(&s2[a][b], &s2[a][b + 1]);

system("cls");

draw2();

break;

}

else continue;

case 27:system("cls"); //ESE返回菜单

menu(); break;

case 48:exit(0); //退出游戏

default:continue;

}

if (s2[1][0] == 2) //到达终点

{

printf("\n\t------------------you are great!--------------------\n\n\n");

s2[1][0] = 3; //重置起点及终点

s2[18][19] = 2;

menu();

}

} while (1);

}

同过检验键盘按键来做出回应调整

4、不足

没能体现出人与机器同时对战的激烈，刺激感觉。

5、心得体会

机器在解迷宫解密码等，能用穷举法解决的问题上相对于人类有天然巨大的优势。我们以后在设计算法时应该想到这一点，并充分发挥此特性。