#include<stdio.h>

#include<conio.h>

#include<windows.h>

#include<stdlib.h>

#include<time.h>

#define Key\_Up 'w' // 向ò上?方?向ò键ü 　?　?

#define Key\_Down 's' // 向ò下?方?向ò键ü

#define Key\_Right 'd' // 向ò右?方?向ò键ü

#define Key\_Left 'a' // 向ò左哩?方?向ò键ü

#define Key\_Space ' '

#define R 1 //向ò右?的?状痢?态?量?

#define L 2 //向ò左哩?的?状痢?态?量?

#define U 3 //向ò上?的?状痢?态?量?

#define D 4 //向ò下?的?状痢?态?量?

typedef struct node

{

int x;

int y;

struct node\*next;

}snake;

//////////全?局?变?量?

int score=0;

int endgamestatus=0;

int food\_x,food\_y;

snake\*head;//蛇?的?头?结á点?

snake\*p;//遍括?历え?蛇?身Θ?用?的?指?针?

int status=R;//蛇?状痢?态?变?量?

int key;

/////////

void endgame();//退?出?游?戏·函ˉ数簓

void Pos(int x,int y);//光a标括?定¨位?函ˉ数簓

void crosswall();//判D断?蛇?是?否?撞?到?墙?壁括?

void Creat\_Food();//生Θ?成é食?物?

int Bit\_Self();//判D断?蛇?头?是?否?与?蛇?身Θ?有瓺接ó触洌?

void Crat\_Map();//生Θ?成é地?图?

void Snake\_Moving();//蛇?身Θ?移?动ˉ

void gamecircle();// 游?戏·循-环·

void pause();//游?戏·暂Y停?

////////

void Pos(int x,int y)

{

COORD pos;

HANDLE hOutput;

pos.X=x;

pos.Y=y;

hOutput=GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleCursorPosition(hOutput,pos);

}

int Bit\_Self()

{

p=head->next;

while(p)

{

if(p->x==head->x&&p->y==head->y)

return 1;

p=p->next;

}

return 0;

}

void Creat\_Map()

{

int i,j;

for(i=0;i<=54;i++)

for(j=0;j<=26;j++)

{

Pos(i,0);

printf("@");

Pos(i,26);

printf("@");

Pos(0,j);

printf("@");

Pos(54,j);

printf("@");

}

}

void crosswall()

{

if(head->x==0||head->y==0||head->x==54||head->y==26)

{

endgamestatus=1;

endgame();

}

}

void Creat\_Food()

{

srand(time(NULL));

food\_x=rand()%50+2;

food\_y=rand()%24+2;

Pos(food\_x,food\_y);

printf("@");

}

void Init\_Snake()

{

int i;

snake\*tail;

head=(snake\*)malloc(sizeof(snake));

head->x=25;

head->y=5;

head->next=NULL;

for(i=1;i<4;i++)

{

tail=(snake\*)malloc(sizeof(snake));

tail->x=25+i\*1;

tail->y=5;

tail->next=head;

head=tail;

}

while(tail)

{

Pos(tail->x,tail->y);

printf("@");

tail=tail->next;

}

}

void Snake\_Moving()

{

snake\*newhead;

newhead=(snake\*)malloc(sizeof(snake));

crosswall();

if(Bit\_Self())

{

endgamestatus=2;

endgame();

}

if(status==R)//向ò右?走?

{

if(head->x==food\_x&&head->y==food\_y)

{

score=score+10;

newhead->x=head->x+1;

newhead->y=head->y;

newhead->next=head;

head=newhead;

p=head;

while(p)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Creat\_Food();

}

else

{

newhead->x=head->x+1;

newhead->y=head->y;

newhead->next=head;

head=newhead;

p=head;

while(p->next->next)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Pos(p->next->x,p->next-> y);

printf(" ");

free(p->next);

p->next=NULL;

}

}

if(status==L)

{

if(head->x==food\_x&&head->y==food\_y)

{

score=score+10;

newhead->x=head->x-1;

newhead->y=head->y;

newhead->next=head;

head=newhead;

p=head;

while(p)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Creat\_Food();

}

else

{

newhead->x=head->x-1;

newhead->y=head->y;

newhead->next=head;

head=newhead;

p=head;

while(p->next->next)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Pos(p->next->x,p->next-> y);

printf(" ");

free(p->next);

p->next=NULL;

}

}

if(status==D)

{

if(head->x==food\_x&&head->y==food\_y)

{

score=score+10;

newhead->x=head->x+1;

newhead->y=head->y;

newhead->next=head;

head=newhead;

p=head;

while(p)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Creat\_Food();

}

else

{

newhead->x=head->x;

newhead->y=head->y+1;

newhead->next=head;

head=newhead;

p=head;

while(p->next->next)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Pos(p->next->x,p->next-> y);

printf(" ");

free(p->next);

p->next=NULL;

}

}

if(status==U)

{

if(head->x==food\_x&&head->y==food\_y)

{

score=score+10;

newhead->x=head->x;

newhead->y=head->y-1;

newhead->next=head;

head=newhead;

p=head;

while(p)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Creat\_Food();

}

else

{

newhead->x=head->x;

newhead->y=head->y-1;

newhead->next=head;

head=newhead;

p=head;

while(p->next->next)

{

Pos(p->x,p->y);

printf("@");

p=p->next;

}

Pos(p->next->x,p->next-> y);

printf(" ");

free(p->next);

p->next=NULL;

}

}

}

void gamecircle()

{

Pos(57,4);

printf("操ù作痢?说μ明÷");

Pos(57,5);

printf("w a s d分?别纄对?应畖上? 左哩?下? 右?");

Pos(57,6);

printf("按恪?空?格?键ü暂Y停?);

while(1)

{

Pos(57,7);

printf("游?戏·分?数簓:%d",score);

if(kbhit())

key=getch();

switch(key)

{

case Key\_Right:

if(status!=L)

status=R;

break;

case Key\_Left:

if(status!=R)

status=L;

break;

case Key\_Up:

if(status!=D)

status=U;

break;

case Key\_Down:

if(status!=U)

status=D;

break;

case Key\_Space:

pause();

break;

default:

break;

}

Sleep(300);

Snake\_Moving();

}

}

void pause()

{

while(1)

{

if(key=getch()==' ')

break;

}

}

void endgame()

{

system("cls");

Pos(27,13);

if(endgamestatus==1)

printf("您ú撞?到?墙?了?");

if(endgamestatus==2)

printf("您ú咬啊?到?了?自?己o");

Pos(27,14);

printf("您ú的?得?分?为a%d",score);

exit(0);

}

void welcome()

{

Pos(27,13);

printf("欢?迎?来ぁ?到?贪?吃?蛇?游?戏·");

system("pause");

system("cls");

Pos(50,9);

printf("欢?迎?大洙?家ò对?源′代洙?码?进?行D修T改?");

Pos(50,11);

printf("开a发ぁ?出?更ü多à好?玩?的?玩?法ぁ?);

Pos(50,12);

system("pause");

system("cls");

}

int main()

{

welcome();

Creat\_Map();

Creat\_Food();

Init\_Snake();

gamecircle();

return 0;

}