

C语言写个贪吃蛇游戏

脚本之家 昨天

以下文章来源于嵌入式Linux，作者写代码的篮球球痴



嵌入式Linux

嵌入式交流 Android linux内核 linux驱动 单片机 程序人生



脚本之家
你与百万开发者在一起



本文经授权转自公众号 嵌入式Linux (ID: Linux-c-world)

如若转载请联系原公众号



贪吃蛇是个非常经典的游戏，用C语言来实现也是一个好玩的事情。这个游戏我写完后放在知乎，竟然点赞的人数超级多。我觉得大家喜欢，一个方面是因为写得简单，大家都能看得懂，一个可扩展性还是非常强的。

我试了说一下这个代码核心的三个函数

```
menu();
setup();
draw();
```

menu用来设置菜单，也就是我们一运行看到的那个。setup用来设置参数，我们需要设置高度和宽度，还有分数，食物的位置。draw也就是画，也就是画整个画面。

还有一个枚举类型这个结构体用来设置蛇的几个状态，我觉得这个也是面向对象编程的一个思想，把蛇的状态都封装成一个枚举类型。

```
typedef enum
{
    STOP = 0,
    LEFT,
    RIGHT,
    UP,
    DOWN
}Direction;
```

还有

```
/*判断贪吃蛇的长度*/
void logic()
```

这个函数，这个函数应该是整个贪吃蛇的精髓了，要理解代码怎么把蛇给连接起来。用了点巧妙的东西。

来看这里面的关键代码

```
/*把上一个位置记下*/
int lastX = tailX[0];
int lastY = tailY[0];
int last2X, last2Y;
/*重新获取当前的位置*/
tailX[0]=x;
tailY[0]=y;
int i=0;
/*遍历整条蛇的长度 把 0 的位置空出来，其余蛇的位置往后面的空间移动*/
for(i=1; i<ntail;i++)
{
    last2X = tailX[i];
    last2Y = tailY[i];
    tailX[i]=lastX;
    tailY[i]=lastY;
    lastX = last2X;
    lastY = last2Y;
}
```

lastX lastY 用来存上一次的蛇头的位置。后面的 for 循环，通过tail 蛇的长度，把蛇上个状态给保存到数组tailX tailY里面。

完整代码

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <Windows.h>
#include <time.h>
#include <conio.h> /*键盘输入获取*/

bool gameOver;
bool stop = false;
bool hit = false;
/*游戏的边框大小*/
const int width = 50;
const int height = 20;
/*蛇的坐标，食物的坐标还有分数*/
int x,y,fruitX,fruitY,score;
/*蛇每个点的坐标*/
int tailX[200],tailY[200];
/*蛇的默认长度*/
int ntail=3;

typedef enum
{
    STOP = 0,
    LEFT,
    RIGHT,
    UP,
    DOWN
}Direction;

Direction Dir;

/*开始菜单*/
void menu()
{
    int a;
    printf("-----\n");
    printf("|                贪吃蛇游戏                |\n");
    printf("|                1) 新游戏                |\n");
    printf("|                2) 开始边界                |\n");
    printf("|                3) 退出游戏                |\n");
    printf("-----\n");
    printf("---->请输入你的选择:");
    scanf("%d", &a);
}

/*初始化状态*/
void setup()
{
    gameOver = false;
    /*根据当前时间设置“随机数种子”*/
```

```
srand(time(NULL));
Dir = STOP;

/*贪吃蛇的位置,固定在中间*/
x= width/2;
y= height/2;
/*食物的位置，位置是随机的*/
fruitX = rand()%width;
fruitY = rand()%height;
score = 0;
}
/*绘制界面*/
void draw()
{
    if(stop == true)
    {
        return;
    }
    system("cls");/*清除屏幕*/
    printf("分数:%d",score);
    printf("\n");

    /*第一行*/
    int i;
    for(i= 0 ;i<width+1;i++)
    {
        printf("-");
    }
    printf("\n");

    /*画中间的画面*/
    int p;
    for(p= 0 ;p<height;p++)/*高度*/
    {
        int q;
        for(q= 0 ;q<width;q++)/*宽度*/
        {
            /*第一行最后已给字符*/
            if(q==0 || q==width-1)
            {
                printf("|");
            }

            if(p == fruitY && q == fruitX)/*食物的随机坐标*/
            {
                printf("O");
            }

            else
            {
                int k=0;
                bool print = false;
                /*贪吃蛇的长度 默认长度是 3*/
                for(k=0;k<ntail;k++)
                {
                    if(tailX[k]==q && tailY[k]==p)
                    {
                        printf(" ");
                        print = true;
                    }
                }
                /*如果这个位置打印了 * 就不要打印空格了*/
                if(!print)
                {
                    printf(" ");
                }
            }
        }
        printf("\n");
    }

    /*最后一行*/
    int j;
    for(j= 0 ;j<width+1;j++)
    {
        printf("-");
    }
}

/*按键输入控制*/
```

```
void input()
{
    if(_kbhit())
    {
        /*获取键盘的输入字符*/
        switch(_getch())
        {
            case '4':
            case 75:/*左键*/
                Dir = LEFT;
                hit= true;
                break;
            case '8':
            case 72:/*上键*/
                Dir = UP;
                hit= true;
                break;
            case '6':
            case 77:/*右键*/
                Dir = RIGHT;
                hit= true;
                break;
            case '2':
            case 80:/*向下键盘键 */
                Dir = DOWN;
                hit= true;
                break;
            case 'x':
            case 27:/*ESE*/
                gameOver = true;
                break;
            case 32:/*空格 暂停键*/
                stop = !stop;
                break;
        }
    }
    else if(!hit && stop == false)/*如果没有改变方向*/
    {
        x++;
    }
}

/*判断贪吃蛇的长度*/
void logic()
{
    if(stop == true)
    {
        return;
    }

    /*把上一个位置记下*/
    int lastX = tailX[0];
    int lastY = tailY[0];
    int last2X, last2Y;
    /*重新获取当前的位置*/
    tailX[0]=x;
    tailY[0]=y;
    int i=0;
    /*遍历整条蛇的长度 把 0 的位置空出来，其余蛇的位置往后面的空间移动*/
    for(i=1; i<ntail;i++)
    {
        last2X = tailX[i];
        last2Y = tailY[i];
        tailX[i]=lastX;
        tailY[i]=lastY;
        lastX = last2X;
        lastY = last2Y;
    }
    /*根据方向来改变x y 的值*/
    switch(Dir)
    {
        case UP:
            y--;
            break;
        case DOWN:
            y++;
            break;
        case LEFT:
            x--;
            break;
```

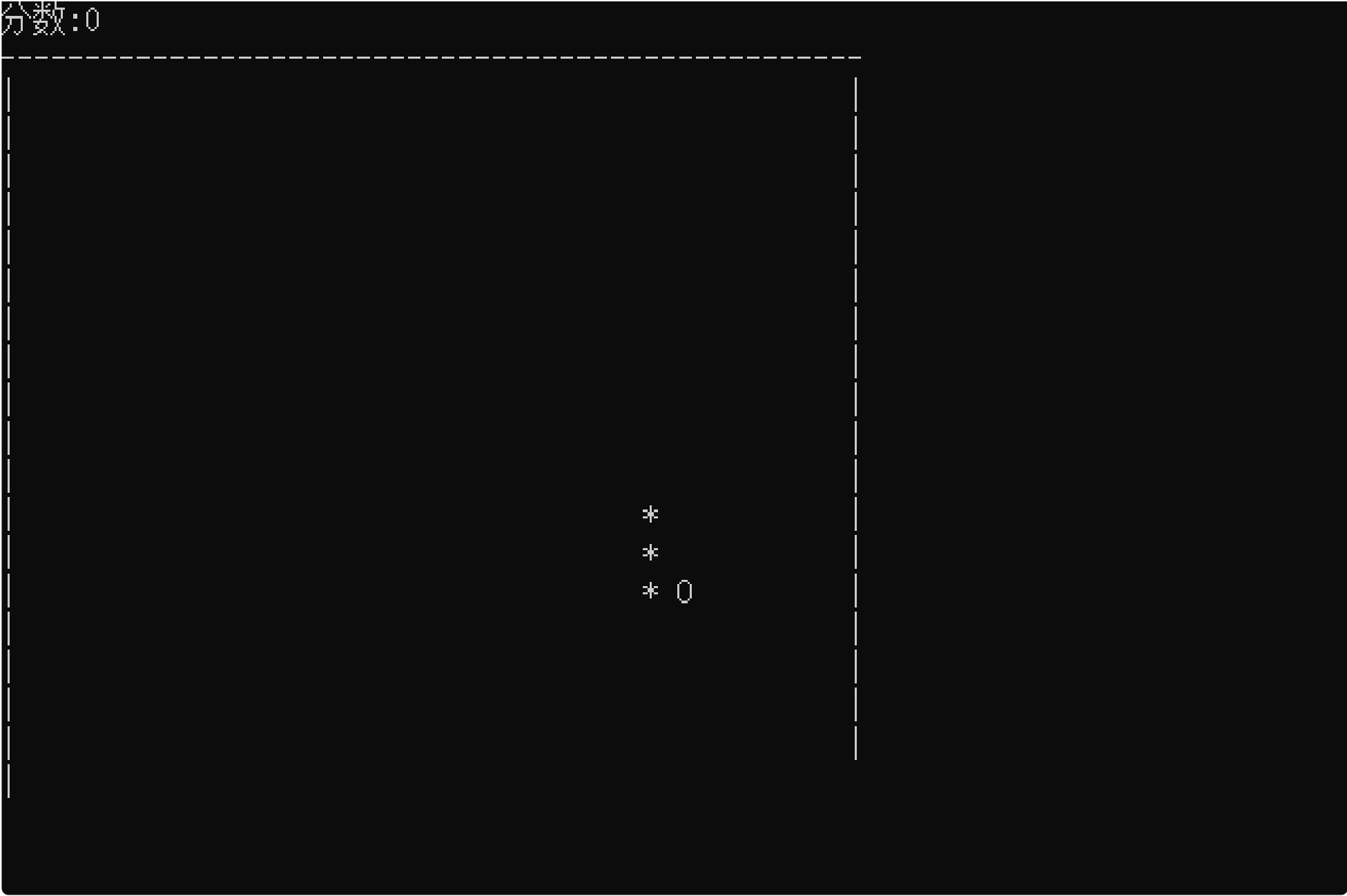
```
        case RIGHT:
            x++;
            break;
    }
    if(x<0 || width<x || y<0 || height<y)
    {
        gameOver = true;
        /*清除屏幕*/
        system("cls");
        printf("-----\n");
        printf("|                                     |\n");
        printf("|                                     |\n");
        printf("|                               游戏结束                               |\n");
        printf("|                                     |\n");
        printf("|                                     |\n");
        printf("-----\n");
    }
    if(x==fruitX && y==fruitY)
    {
        /*吃了一个食物，蛇的长度增加1*/
        ntail++;
        score+=10;
        /*更新下一个食物的位置*/
        fruitX = rand()%width;
        fruitY = rand()%height;
    }
}

int main()
{
    #if 0
        while(1)
        {
            printf("%d\n",_getch());
        }
    #endif
    menu();
    setup();
    draw();
    /*循环画贪吃蛇的界面*/
    while(!gameOver)
    {
        draw();
        input();
        logic();
        Sleep(70);
    }

    return 0;
}
```

上面这段代码直接在Dev C++上面应该是可以运行的，很多人在知乎上私信问我，为什么我的贪吃蛇执行不了呢，可能就是平台不同，少了这个头文件，少了那个头文件，但是你为什么不能跟我一样，用Dev C++呢，轻量级，简单。代码的精髓是什么？我认为精髓一定是思想，不是你写了多少行代码，用了什么高端的IDE。

 C:\Users\weiqifa\Desktop\未命名1.exe



我自认为我的注释已经写得不错了，所以就没有什么好说明的了吧，有不明白的把代码过一下，至于屏幕刷新这个东西，如果只是用时间刷新就会闪屏，所以出现了一个双缓存，把要显示的东西送到一个**buff**里面去，另一个**buff**用来显示，这样就可以保证不会出现闪屏。除了写贪吃蛇，可以用这个方法写其他小程序，挺有意思的。

在知乎上，发起了一个C语言 100 行代码之内实现贪吃蛇的问题。我觉得很不错，里面很多同学的回复都非常赞，特别是叶大神的回复。

<https://www.zhihu.com/question/360814879/answer/1013986215>

- END -

 脚本之家

签到啦

小程序

更多精彩

在公众号后台对话框输入以下**关键词**
查看更多优质内容！

女朋友 | **大数据** | **运维** | **书单** | **算法**
大数据 | **JavaScript** | **Python** | **黑客**
AI | **人工智能** | **5G** | **区块链**
机器学习 | **数学** | **送书**

- [人人都欠微软一个正版？](#)
- [脚本之家粉丝福利，请查看！](#)
- [程序员怒打产品经理，这个需求做不了](#)
- [致敬经典：Linux/UNIX必读书单推荐给你](#)
- [一个故事讲完CPU的工作原理](#)
- [终于有人把 Nginx 说清楚了，图文详解！](#)

