

# L0: 直接运行在硬件上的小游戏 (amgame)

姓名: 殷天润 学号: 171240565

2019 年 3 月 6 日

## 第一部分 主要架构

1. 我的小游戏是贪吃蛇, 用 wsad 进行操作, 如果撞墙或者在吃到第一个食物之后撞自己的身体就会进入 Game Over 页面; 按空格键重新开始游戏.
2. 我的程序主要分为三个部分: font.c, draw.c, game.c. font.c 与 draw.c 借鉴了打字小游戏, 并根据我的代码做了一定的改变。game.c 里面包含了按键, 蛇的状态更新, 蛇的绘画, 食物的创建, 吃食物的判断、是否失败的判断。我对 game.h 也进行了一点改动;
3. 主要的思路是:
  - (a) 将蛇看作是一个结构体数组, 结构体里面的两个属性是 `_snake_x, _snake_y`, 用来确定蛇这部分的位置, 这在 game.h 里面可以看到 (我假设这个游戏得分不超过 100, 因此结构体数组就开了 100);
  - (b) 每一次按键读取后, 获得方向 (`direction_x, direction_y`) 的值;
  - (c) 用获得的方向值来更新蛇头, 除了蛇头之外的部分都只要替换成前面一个就行了, 在替换前记录尾部的位置
  - (d) 把蛇身子画出来, 并且把之前记录的蛇尾给”擦去”
  - (e) 然后考虑食物的产生, 用一个 flag 判断食物有没有吃以及一个随机数生成就可以解决了;
  - (f) 只要在原来的 while 外面再加一个 while, 以及一个判断游戏失败/重新开始的 flag 就可以实现无尽版的贪吃蛇了.
4. game.h 中:

我定义了蛇的结构体, 定义了各种全局变量, 宏定义了颜色;
5. draw.c 中:

`draw_character`, 我把打字小游戏里面的二维数组存储改成了 `draw_rect2` 直接画在屏幕上
6. game.c 中:

- (a) `init_screen()` 是对 ioe 的初始化, 我没有任何修改
- (b) `init_snake()` 对蛇初始化
- (c) `food()` 随机在蛇吃掉食物的时候随机出食物
- (d) `GG()` 用来判断撞墙和撞自己
- (e) `splash()` 将整个界面画成白色
- (f) `_snakemove` 用来通过键盘中获得的信息更新蛇这个数组的信息并且对是否吃掉食物进行了判断
- (g) `_snakedraw` 用来画蛇
- (h) `main` 函数中, 我为了在 `GG` 之后继续运作又加了一个 `while`, 这样在接受到空格键的 `flag` 之后就可以重新开始游戏了。

## 第二部分 遇到的问题

1. 遇到了各种编译以及笔误;
2. 之前遇到了一个食物出现在蛇的身子里面的 bug, 然后我修改了 `eat` 这个 `flag` 的判断, 将仅对于头部进行判断改成了对整个蛇的身子进行判断

## 第三部分 潜在的完善

1. 关于蛇的数组问题, 我只要再加一个平行于 `Game Over` 界面的 `while` 就可以了, 在接受到 `total==100` 的时候用一个 `flag` 就行了 (感觉没啥新意我就没有实现)
2. 关于游戏性的问题, 可以做成两条蛇的
3. 关于 `draw_character`, 可以优化成 `draw_string`, 但是我需求不大, 就没有写这个