

《创新训练课B》课程设计报告

学院： 机电工程与自动化学院

题目： **大鱼吃小鱼双人版**

班级： **自动化4班**

姓名： **吴坤远**

学号： **200320428**

教师： **吴晓军**

上交日期： 2021.10.10

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2021.8.20 | V1.0 | 完成代码框架 | 吴坤远 |
| 2021.8.26 | V1.5 | 实现鱼的成长和随机生成 | 吴坤远 |
| 2021.9.27 | V2.0 | 优化碰撞体积 | 吴坤远 |
|  |  |  |  |

**目 录**

[1 引言 1](#_Toc519846913)

[1.1编写目的 1](#_Toc519846914)

[1.2背景 1](#_Toc519846915)

[1.3定义 1](#_Toc519846916)

[1.4参考资料 1](#_Toc519846917)

[2 任务概述 1](#_Toc519846918)

[3 需求分析 1](#_Toc519846919)

[3.1 用户需求分析 1](#_Toc519846920)

[3.2 运行环境 1](#_Toc519846921)

[4 功能及操作介绍 1](#_Toc519846922)

[4.1 操作 1](#_Toc519846923)

[4.2 功能 1](#_Toc519846924)

[5 系统设计 2](#_Toc519846925)

[5.1 总体架构设计 2](#_Toc519846926)

[5.2 模块分析与设计 2](#_Toc519846927)

[5.3软件结构（流程图） 2](#_Toc519846928)

[6 调试与测试 2](#_Toc519846929)

[6.1调试过程 2](#_Toc519846930)

[6.2测试结果 2](#_Toc519846931)

[7 编程中遇到的问题 2](#_Toc519846932)

[7.1 问题1 2](#_Toc519846933)

[7.2 问题2 2](#_Toc519846934)

[7.3 问题3 2](#_Toc519846935)

[8分析总结与心得体会 2](#_Toc519846936)

# 1 引言

## 1.1编写目的

C语言程序设计课程设计是集中实践性环节之一，是学习完《高级语言程序设计》课程后进行的一次全面的综合练习。要求学生达到熟练掌握C语言的基本知识和技能;基本掌握面向过程程序设计的思想和方法；能够利用所学的基本知识和技能，解决简单的面向过程程序设计问题，从而提高动手编程解决实际问题的能力。

## 1.2背景

游戏是一种基于物质需求满足之上的，在一些特定时间、空间范围内遵循某种特定规则的，追求精神世界需求满足的社会行为方式，但同时这种行为方式也是我们所必需的一种降压减排的方式。

合理适度的游戏允许人类在模拟环境下挑战和克服障碍，可以帮助人类开发智力、锻炼思维和反应能力、训练技能、培养规则意识等。大鱼吃小鱼主要锻炼玩家的判断能力和反应速度。

## 1.3定义

大鱼吃小鱼出自汉刘向《说苑．指武》：“大之伐小，强之伐弱，犹大鱼之吞（吃）小鱼也。”后以“大鱼吃小鱼”比喻大欺小，强凌弱。

在看似风平浪静的大海里，却隐藏着与人类世界相同的危机，海底生物在弱肉强食的竞争下，不但得觅食以填饱肚子，更重要的是学会保护自己。

在大鱼吃小鱼游戏中，玩家要躲避大鱼，吃掉小鱼，实现自我的生存和成长。

## 1.4参考资料

<http://www.4399.com/flash/93398.htm> 大鱼吃小鱼双人版

<https://docs.easyx.cn/zh-cn/intro> easyX使用文档

# 2 任务概述

项目实现了大鱼吃小鱼双人版游戏。

首先有三个主界面：游戏菜单界面、游戏界面、游戏结束界面。在游戏菜单界面内能查看操作说明和游戏介绍，并开始游戏；在游戏界面中，玩家可选择单人或双人，在弱肉强食的海底世界操控角色，躲避大鱼，吃掉小鱼，尽可能的获得更高的分数；游戏结束界面会显示分数，并能重新开始回到游戏菜单。

# 3 需求分析

## 3.1 用户需求分析

首先，游戏界面要提供足够的引导信息，比如操作说明和游戏介绍；然后要有音效和音乐，提供沉浸式体验；最后要调整好游戏的平衡性，既要有难度，又要有趣味。

## 3.2 运行环境

操作系统：windows10

# 4 功能及操作介绍

## 4.1 操作

## ①点击fish\_game.exe进入游戏。

## ②可在菜单界面查看操作说明、游戏介绍，并开始游戏。

③可选择单人模式或双人模式。

④在游戏界面使用WASD和上下左右键进行角色操控。

## 4.2 功能

## ①鱼会在游戏界面左右边界随机生成并横向移动。

②玩家会随着吃掉鱼（分数增加）而变大。

# 5 系统设计

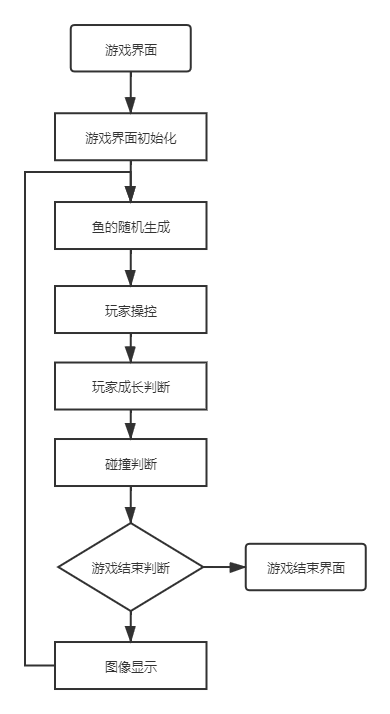
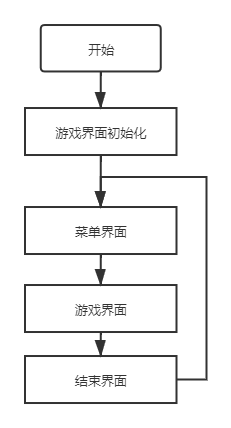
## 5.1 总体架构设计

|  |
| --- |
| int main(void)  {  //游戏窗口初始化  //initgraph(800, 600, SHOWCONSOLE);  initgraph(800, 600);  bgm();  while(1)  {  //菜单  menu();  //游戏界面  game(choice());  //游戏结束界面  gameover();  }  closegraph();  return 0;  } |

## 5.2 模块分析与设计

|  |
| --- |
| void game(int player)  {  //游戏初始化  gameinit(player);    //游戏  BeginBatchDraw();  while(1)  {  //鱼的随机生成  born\_fish();  //玩家的操控  keyboard(0);  if (player == 2)  {  keyboard(1);  }  //玩家成长判断  check\_player(0);  if(player == 2)  {  check\_player(1);  }  //碰撞判断  check\_fish();  count1++;  if (player == 2)  {  count2++;  }  //游戏是否结束  if (check\_gameover())  {  break;  }    //界面  setbkmode(TRANSPARENT);  putimage(0, 0, bg);  settextstyle(50, 0, "Consolas");  settextcolor(RED);  showtext(30, 30, "1P");  showtext(230, 30, "SCORE:");  if (player == 2)  {  showtext(430, 30, "2P");  showtext(630, 30, "SCORE:");  }  //显示鱼  show\_fish();  //显示玩家  show\_player(0);  if (player == 2)  {  show\_player(1);  }  //显示积分  show\_score(0);  if (player == 2)  {  show\_score(1);  }  FlushBatchDraw();  Sleep(30);  cleardevice();  }  EndBatchDraw();  } |
| //鱼的结构体  struct fish  {  int x;  int y;  int width;  int height;  int check\_width;  int check\_height;  int speed;  int is\_alive;  int score;  IMAGE\* bg;  IMAGE\* self;  };  typedef struct fish FISH;  FISH fish[fish\_number]; |
| struct player  {  int x;  int y;  int width;  int height;  int direction; // 0:--> 1:<--  int speed;  int life;  char Life[10];  int score;  char Score[10];  IMAGE\* bg;  IMAGE\* self;  };  typedef struct player PLAYER;  PLAYER player[2]; |

## 5.3软件结构（流程图）



# 6 调试与测试

## 6.1调试过程

从玩家角度进行试玩。

## 6.2测试结果（包括界面，数据，结果）

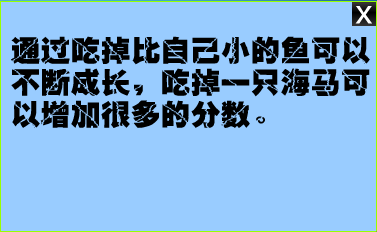
游戏菜单



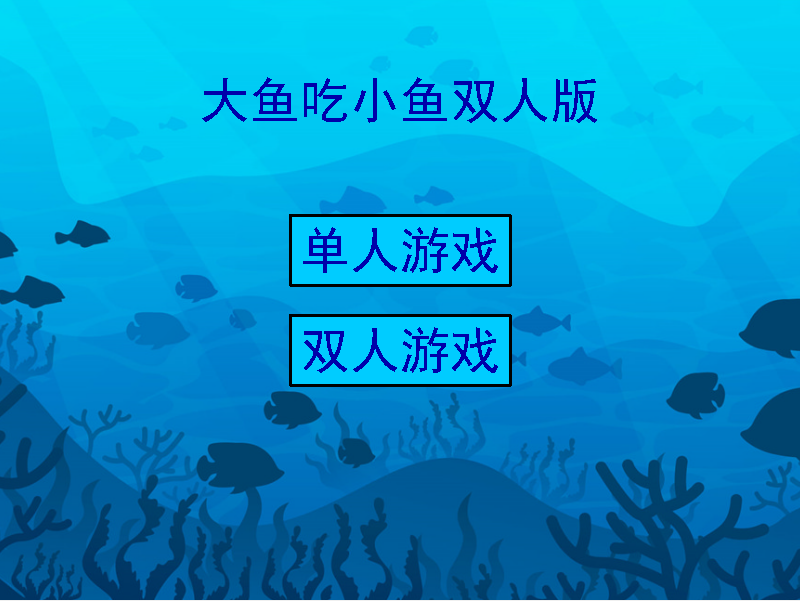
操作说明



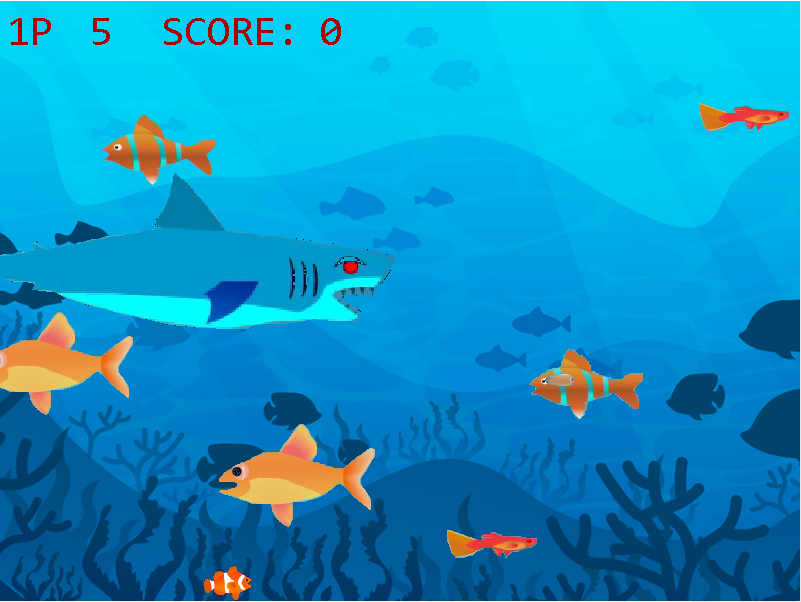
游戏介绍



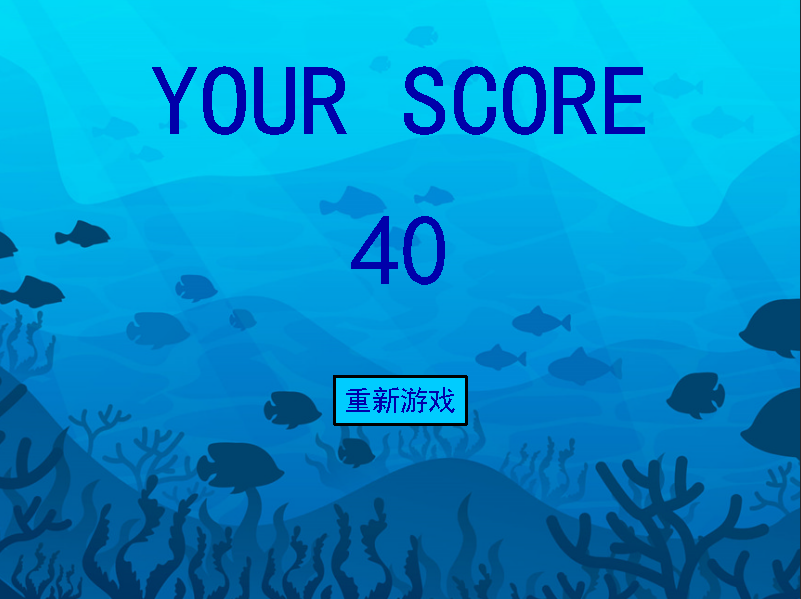
单人模式或双人模式



游戏界面



游戏结束界面



# 7 编程中遇到的问题

## 7.1 问题1

在设计菜单界面时，为了实现将鼠标移动到按钮上按钮闪动特效纠结了很久逻辑设计。

最后找到了一个逻辑方案：先判断鼠标位置，不在按钮范围就显示默认主界面；然后依次判断在哪个按钮范围，若按下左键则运行按钮对应程序，若未按下就闪动。该游戏所有界面的按钮都设置了闪动特效。

## 7.2 问题2

在设计碰撞判断时，测试发现玩家一旦碰到大鱼会瞬间死亡，为了避免这种情况发生，我设计了两个全局变量：初始化count=0，当玩家与大鱼发生体积碰撞后，count++，直到到达一个阈值后才判定减去一格生命值。

## 7.3 问题3

使用new和delete不规范导致出现内存错误，发现后把所有图像存储空间都放在全局变量中new。

# 8分析总结与心得体会

通过本次C语言项目实践，我对C语言的应用和项目有了更深的了解。