

课程设计报告

选题名称：私人火箭运营模拟

完成日期：2020 年 2 月 1 日

电子信息学院

目录

1 课程设计概述	1
1.1 课程设计目的	1
1.2 课程设计的教学内容与要求	1
1.3 课程设计的评价	2
2 问题描述	2
3 功能需求	3
4 系统分析	3
4.1 系统功能模块	3
4.2 数据的存储	3
4.3 记录属性分析	4
5 系统设计	5
5.1 数据结构设计	5
5.2 函数设计	6
5.3 文件清单	8
5.4 部分功能说明	8
6 系统运行界面	10
6.1 系统运行主界面	10
6.2 开发者工具运行主界面	11
6.3 自动初始化运行主界面	12
7 系统测试	12
8 系统完善	13
8.1 基础数据变动后容易丢失	14
8.2 用户使用时界面跳转过快导致没有浏览完信息	14
8.3 输入非法数据后容易造成死循环	14
参考文献	14

1 课程设计概述

1.1 课程设计目的

高级语言程序设计是计算机类专业学生必须掌握的专业基础课，课程以影响深远的、传统的、面向测试过程的优秀编程语言 C 语言为基础，学习程序设计的基本概念和方法，通过本门课程学习，学生应掌握高级语言的基本原理，熟练掌握程序设计的基础知识、基本概念和程序设计的思想和编程技巧。

高级语言程序设计是一门实践性很强的课程，课程的实践方式有两种，一是课程内的实验，即向学生布置针对课堂教学内容的小型练习题，由学生独立进行程序设计与上机实现；二是综合性的课程设计，更强调知识的整合、问题分析与求解能力及团队合作能力的培养。因而，课程设计更能培养学生综合运用所学理论知识解决复杂问题的实践能力、研究性的学习能力和团队精神。

课程设计不仅仅是以实现相应的程序、完成指定的功能为目标，更重要的是在完成课程设计的过程中逐步培养今后从事专业工作所需的各种能力与素质，其中较为重要的是设计文档的写作能力。因此，在课程设计实施中，不仅需要完成程序并进行调试，还需撰写相应的课程设计报告，课程设计报告不仅是对课程设计的总结，也是对课题的设计、开发、软件的使用文档等写作能力的初步培训。同时，课程设计的考核过程中有项重要环节，即课程设计答辩，以课程设计项目组为对象，通过学生自述、教师提问与学生回答等方式训练学生的综合表达能力。

1.2 课程设计的教学内容与要求

高级语言程序课程设计不仅是对程序设计能力的综合锻炼，更是对团队合作、软件开发与项目管理过程的训练。因此，课程设计根据题目的难度由小组合作完成，每个小组 3~4 人。整个课程设计分为以下几个阶段。

(1) 选题与开题：题目可来自教师指定的参考题目，也可自由选题，特别鼓励有创新性的题目或在已知题目的基础上进行创新。

(2) 系统设计：系统设计的任务是对所确定的题目从问题需求、数据结构、程序结构、难点及关键技术等方面进行分析，形成系统的设计方案，并进行详细的分工。

(3) 系统编码实现：小组成员根据初步的系统设计方案，对系统进行编程实现。

(4) 系统测试：在各程序模块编码完成并集成后，对整个系统进行测试。

(5) 撰写课程设计报告：完成文档整理，按照要求完成课程设计报告。

(6) 系统评价与验收：通过小组答辩的形式对课程设计过程进行评价与验收。

1.3 课程设计的评价

参加本课程设计的学生，应当认真完成本课程设计的全部过程，并以最终课程设计成果来证明其独立完成各种实际任务的能力，从而反映出理解和运用本课程知识的水平和能力，其具体要求如下：

- (1) 数据组织方面尽量用到数组、链表、结构体、文件等内容；
- (2) 程序结构方面做到函数调用，采用指针运算；
- (3) 代码编写规范，有充分的注释，具有良好的编程习惯；
- (4) 程序须有一定的健壮性和必要的提示信息，考虑问题的多种可能和边界数据；
- (5) 课程设计完成须进行答辩，提交课程设计报告电子稿、装订的打印稿；

课程设计报告内容包括以下几个方面：

课题的总体设计 和算法分析；

- ✧ 程序流程图、函数说明、文件说明；
- ✧ 源程序代码清单；
- ✧ 测试数据和测试过程记录；
- ✧ 遇到的问题及解决方法分析，包括对课题解决方案的设想和改进意见等；
- ✧ 课程设计小结；

2 问题描述

研究月球的各种项目有助于推进人类对太空环境的认知和进一步探索。

近日，中国空间技术研究院研制的嫦娥五号月球探测器成功圆满完成了探月任务。这是至今中国航天最复杂、难度最大的任务之一，将实现中国首次月球无人采样返回，助力深化月球成因和演化历史等科学研究，对我国航天领域及各类前沿科技领域具有重大意义。

利用此例作为模板，本课题设计将围绕航天火箭发射进行展开，用 C 语言实现数据的处理、火箭的购买、火箭的发射、收益分成和开发者模式的基本操作，支持数据的写入，删除，查询，修改和文件的读写操作。程序中涉及数据块和文件指针等操作，包括结构体数据的定义，宏定义，以及对文件的覆盖、创建、定位等功能。

背景将参考美国太空运输公司——SpaceX，通过现实中的例子来对课题进行进一步的渲染和丰富。有助于强化学生对创新课题挑战的积极性，开拓眼界，紧

跟时事，了解中国在世界上的飞跃发展。

3 功能需求

系统要求实现私人火箭运营模拟的基本功能，包括数据的写入、修改、删除、查找等。对于数据信息而言，主要包括基础数据结构和火箭数据结构两类，系统达到的功能如下：

- (1) 以菜单的方式工作
- (2) 基础数据的渐进性查询
- (3) 仓库内火箭的查询（数据的查找）
- (4) 商店购买并提示结果（数据的写入）
- (5) 火箭发射（数据的查找和删除）
- (6) 收益计算并更新数据（数据的修改）
- (7) 初始化所有数据（文件和数据的新建覆盖）
- (8) 控制动画的开启与关闭
- (9) 清除仓库数据（文件的新建覆盖）

4 系统分析

4.1 系统功能模块

按系统所需实现的功能要求，功能模块如图 1 所示。

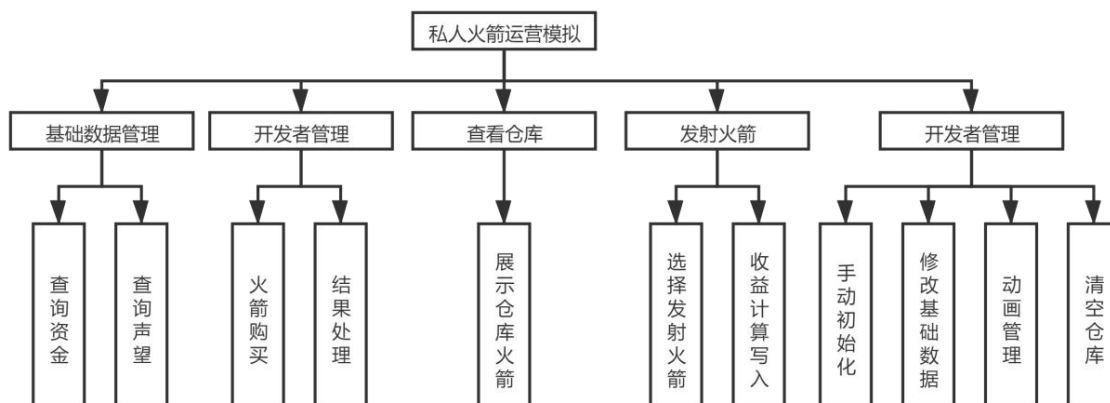


图 1

4.2 数据的存储

私人火箭运营模拟系统所涉及的信息以文件形式存储在计算机磁盘上，为有利于一条基础数据或火箭数据整体的读写，文件类型为二进制，二进制文件中数

据项存储示意图如图 2 所示，图中每个矩形块为一条记录，矩形块的长度相等，即表示每条数据信息占用的存储空间相等。

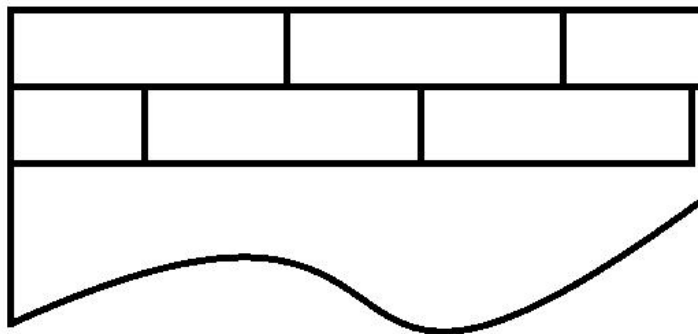


图 2

4.3 记录属性分析

私人火箭运营模拟系统的数据信息主要有两类，即基础数据信息和火箭商品信息，其记录属性如下。

(1) 基础数据信息

- ✧ 公司资金：公司内部总资产，定义初始资金为 10000(单位：万元)，为了简化收入和支出，不细化购买火箭的步骤（例如购买发射器、燃料、科学经费等），不细化回收火箭后的收入问题（样品质量，样品类型等）。
- ✧ 公司声望：公司整体名誉的衡量指标，定义初始声望为 1，取值范围为： $[1, +\infty]$ 为了简化声望变化过程，声望值只会增加而不会减少，增长方式只能通过火箭发射的方式来获得。

(2) 火箭商品信息

- ✧ 火箭类型：火箭类型名称，用来区别火箭的发射方式，取值范围为{单级，星链}。
- ✧ 商品名：火箭商品名称，仅用来区别火箭之间的区别，取值范围为{嫦娥 7 号，嫦娥 8 号，嫦娥 S 号，C 级星，B 级星，A 级星}。
- ✧ 购买价格：火箭商品价格，影响资金数目的途径之一，取值范围为{800, 1200, 2700, 4000, 7500, 14000}。
- ✧ 飞行距离：火箭的飞行距离，用于计算收益，为了方便计算，火箭飞行距离全部为固定最大值，取值范围{1.00, 2.00, 0.30, 0.50, 3.00, 0.75}。
- ✧ 商品编号：火箭的商品编号，取值范围{1, 2, 3, 4, 5, 6}。

(3) 附加属性

系统除描述基础数据和火箭数据的基本信息外，还需达到信息记录的查找修改的功能，于是在基础数据信息和火箭商品信息上附加如下属性。

- ✧ 基础数据附加信息

- 初始化标志：程序是否初始化的判别标志，定义初始值为 1，取值范围为{0,1}，其中当初始值为 1 的时候代表程序已经初始化，初始值为 0 的时候代表未初始化。
- 动画开关标志：程序的动画是否开启标志，定义初始值为 1，取值范围为{0,1}，其中当初始值为 1 的时候代表程序动画开关为开，初始值为 0 的时候代表程序动画开关为关。

✧ 火箭商品附加信息

- 有效状态：当火箭发射完毕后，需要及时在信息中删除，删除某条特定信息是在原有的基础上将有效标记置为“无效状态”，而不是物理上的删除，用有效状态来判断火箭是否有，定义初始值为 1，取值范围为{0,1}，其中当初始值为 1 的时候代表火箭有效，初始值为 0 的时候代表火箭无效。



5 系统设计

5.1 数据结构设计

系统处理的数据主要是火箭和基础数据，从系统分析来看，两者数据结构可用 C 语言结构体的形式来表示。

(1) 基础数据结构

```
struct Companydata //基础数据
{
    int flag; //初始化标志
    int Capital; //资金
    double Prestige; //声望
    int Animation; //动画开关
}Inidata={1,10000,1.000,1};
```

(2) 火箭数据结构

```
struct Singlestagerocket //火箭基本信息
{
    char Rockettype[10]; //火箭类型
    char Tradename[10]; //商品名
```

```
int Price; //购买价格

double Distance; //飞行距离

int number; //商品编号

int IsAvailable; /有效状态

}Rocket[6]={

    {"单级","嫦娥 7 号",4000,1.00,1,1},

    {"单级","嫦娥 8 号",7500,2.00,2,1},

    {"星链","C 级星",800,0.300,3,1},

    {"星链","B 级星",1200,0.500,4,1},

    {"单级","嫦娥 S 号",14000,3.00,5,1},

    {"星链","A 级星",2700,0.75,6,1}

};
```

5.2 函数设计

私人火箭运营模拟系统采用结构化程序设计思想，程序中除了主函数外，共设计了 32 个函数，函数原型及功能说明如表 1 所示。

表 1 函数原型及功能说明

序 号	函 数 原 型	功 能 说 明
1	int MainMenu()	一级菜单，返回用户选择值
2	void CreateSubMenu1(int coin,double Prestige)	二级菜单，查询资金声望界面
3	int CreateSubMenu2(int coin,int Prestige)	二级菜单，购买火箭商店，返回用户选择值
4	int CountRocketSingle()	返回仓库文件中已有单支火箭数量
5	int CountRocketStar()	返回仓库文件中已有星链火箭数量
6	int CountRocket()	返回仓库文件中已有所有类型火箭数量
7	void ShowRocketofwarehouse()	展示仓库文件中已有所有类型火箭
8	void Rocketstorage(int i)	写入对应编号的火箭数据至仓库文件
9	void Launchsettlement(double *Prestige,int *Capital,double s,int Price)	结算界面
10	double Percentagerandomnumber(int down,int up)	百分随机数生成函数，down 和 up 为百分数形式的上下限，返回随机生成的百分数

11	int Sixdigitrandomnumber()	六位随机数生成函数, 返回随机生成的六位随机数
12	int Incomeconversion(double s, double *Prestige, int Price, int *Capital)	飞行距离转换收益函数, 返回收益值
13	int WarehouseselectionSingle()	二级菜单, 发射单支火箭展示仓库状态, 并返回用户选择值
14	int WarehouseselectioStar(int n[])	二级菜单, 发射星链火箭展示仓库状态, 并返回选择的数量
15	int IsRocketExistSingle (int Rocketnumber)	查询指定编号的单支火箭在文件是否存在, 若存在返回 1, 否则返回 0
16	int IsRocketExistStar (int Rocketnumber)	查询指定编号的星链火箭在文件是否存在, 若存在返回 1, 否则返回 0
17	int GetRocketstoragelocation(int number)	查询指定编号的火箭, 返回其存放的位置
18	void DelRocket(int n)	删除指定位置的火箭
19	int Initializationdetection()	初始化检测, 若已初始化返回 1, 否则返回 0
20	void Initializationdata()	初始化基础数据文件
21	void Emptywarehouse()	初始化仓库文件
22	void Savebasicdata(int Capital, double Prestige, int Animation)	保存当前基础数据至基础数据文件
23	int FindCapital()	返回基础数据文件的资金值
24	int FindPrestige()	返回基础数据文件的声望值
25	int FindAnimation()	返回基础数据文件的动画开关值, 若开返回 1, 否则返回 0
26	int InsMenu()	二级菜单, 设置菜单, 返回用户选择值
27	int LaunchCSnMenu()	二级菜单, 发射类别选择菜单, 返回用户选择值
28	void ReviseCandP(int *Capital, double *Prestige)	自定义修改基础数据文件的资金或声望
29	void Rocketanimation(int x)	动画播放函数
30	void Rocketanimation(int x)	火箭动画播放函数
31	void Initializeanimation()	第一次初始化播放函数
32	void InitializeanimationAgain()	第二次初始化播放函数

5.3 文件清单

在表 1 中的 32 个函数与主函数，以及函数声明，数据结构定义，数据存储二进制文件共 6 个磁盘文件，程序文件清单如表 2 所示。

表 2 程序文件清单

序 号	文 件 名	文 件 类 型	文 件 说 明
1	menu.c	C 文件	存储程序的主函数
2	head.h	头文件	存储程序的宏定义，结构体，声明所有自定义函数
3	func.h	头文件	存储程序所有的自定义函数
4	Animation.h	头文件	存储程序所有的动画函数
5	Basicdata.bat	二进制数据文件	存储基础数据
6	Warehouse.bat	二进制数据文件	存储火箭数据

5.4 部分功能说明

本节将选择程序中的部分功能加以分析。

(1) 主菜单选择

按提示输入一个选择值，系统会判断这个值是否合法，即是否为预设的几种选择，如果选择值为非退出值，那么菜单函数将记录下此值，并执行对应的操作，之后继续循环，再次输入新的数据进行后续操作，如果选择值为退出值，那么首先保存一次所有的数据，之后退出循环，关闭程序，保存新纪录至文件内部。

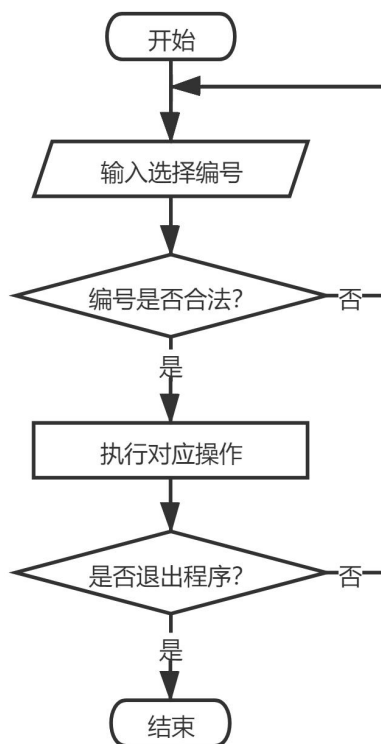


图 3 主菜单选择

(2) 自动初始化检测

系统将会在程序运行之前进行一次初始化检测，系统调用初始化检测函数，检测基础数据文件是否不存在，如果数据文件不存在，系统则会在根目录下新建覆盖两个数据文件，并播放初始化演示动画，进而再进行后续操作。

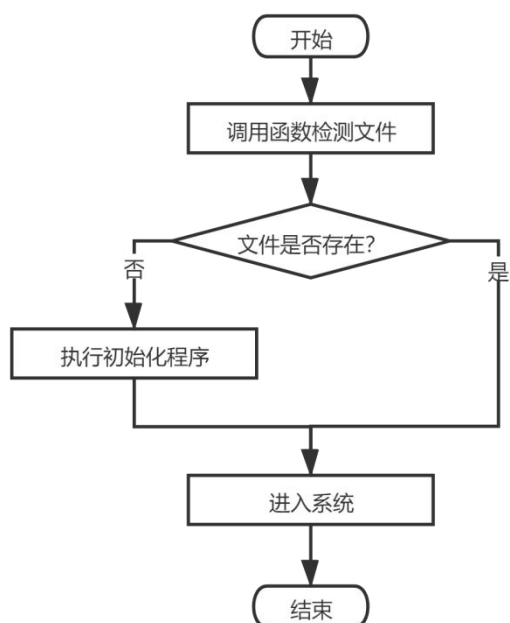


图 4 自动初始化检测

6 系统运行界面

6.1 系统运行主界面

程序运行显示“嫦娥 X 系统”的主菜单界面，等待用户选择执行的菜单项目，如图 5 所示。若输入的选择值不合法则要求重新输入，如图 6 所示。



图 5 “嫦娥 X 系统”的主菜单界面

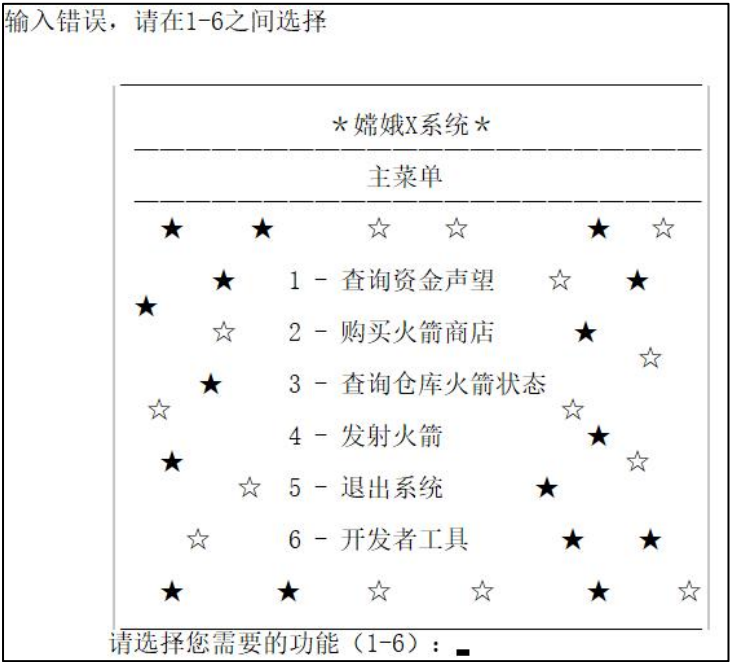


图 6 输入不合法数据的主菜单界面

6.2 开发者工具运行主界面

程序运行显示“嫦娥 X 系统”的开发者工具界面，等待用户选择执行的菜单项目，如图 7 所示。若输入的选择值不合法则要求重新输入，如图 8 所示。

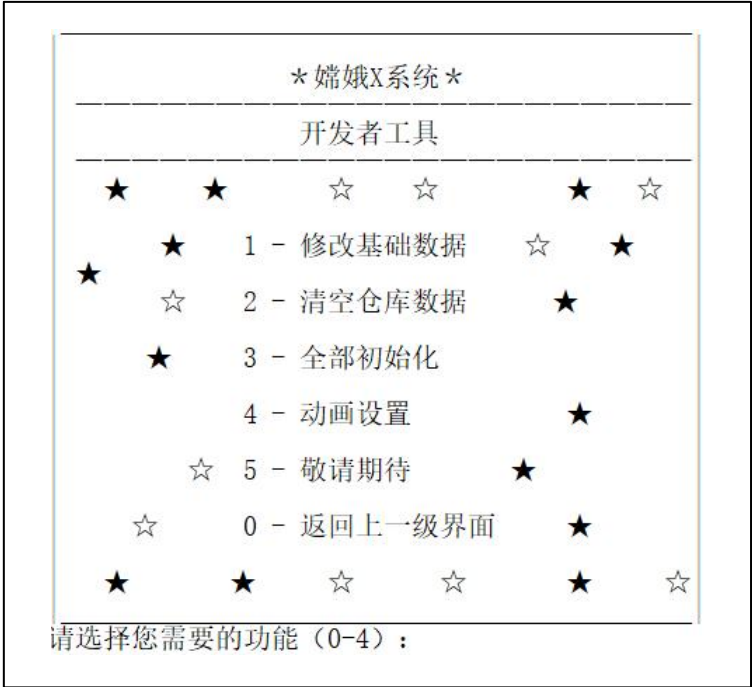


图 7 “嫦娥 X 系统”的开发者工具界面

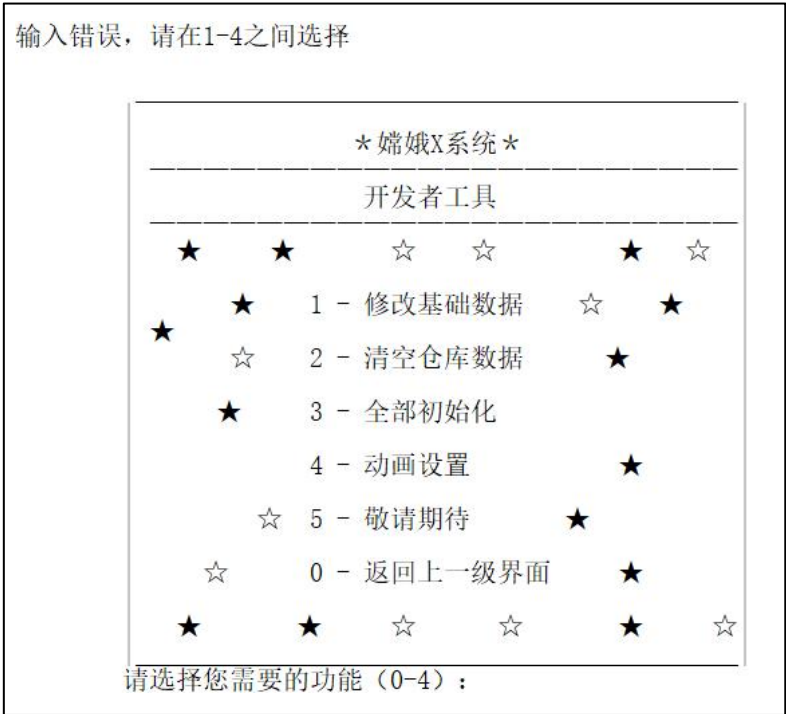


图 8 输入不合法数据的开发者工具界面

6.3 自动初始化运行主界面

程序运行前检测数据文件是否存在，如果不存在则新建覆盖数据文件，并播放初始化动画，如图 9 所示。

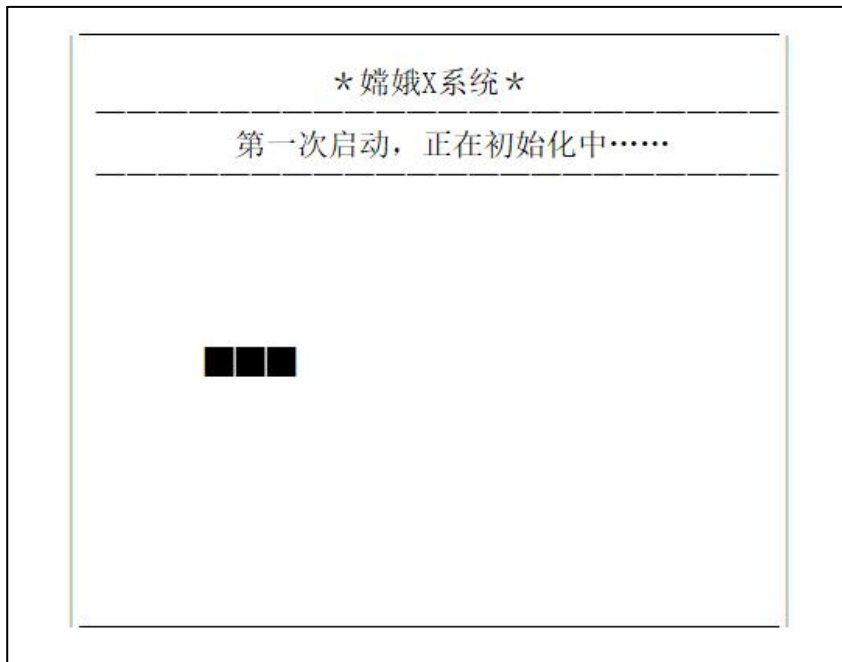


图 9 自动初始化检测界面

7 系统测试

测试是程序设计中至关重要的步骤，是保证软件功能的正确实现及满足用户需求的重要环节。测试之前首先要制订测试方案，包括待测试功能、测试数据及预期结果等。测试方案主要有两种方法，即黑盒测试和白盒测试。

黑盒测试也称功能测试，它是通过测试来检测每个功能是否都能正常使用。在测试中，把程序视为个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息。黑盒测试着眼于程序外部结构，不考虑内部逻辑结构，主要针对软件界面和软件功能进行测试。

白盒测试也称结构测试或逻辑驱动测试，它是按照程序内部的结构测试程序，通过测试来检测产品内部动作是否按照设计规格说明书的规定正常进行，检验程序中的每条通路是否都能按预定要求正确工作。白盒测试是把测试对象视为一个打开的盒子，测试人员依据程序内部逻辑结构相关信息，设计或选择测试用例，对程序所有逻辑路径进行测试，通过在不同点检查程序的状态，确定实际的状态是否与预期的状态一致。

为测试系统各项功能，应预先设计好测试用例并预测相应结果。如表所示列

出了设计的部分测试用例，以及测试程序功能是否正确。

表 3 系统功能测试表

功能名称	测试数据	预期结果	测试结果	测试结论
主菜单项	1	若输入的选择值与菜单选项一致则执行菜单功能，否则提示错误	跳转修改基础数据界面	通过
	2		跳转清空仓库确认界面	
	3		跳转手动初始化界面	
	4		跳转火箭动画设置界面	
	5		输入错误，重新进行输入	
	0		返回主菜单界面	
	c		输入错误，重新进行输入	
	R		输入错误，重新进行输入	
	_3		输入错误，重新进行输入	
	*		输入错误，重新进行输入	
	-		输入错误，重新进行输入	
	34		输入错误，重新进行输入	
	545		输入错误，重新进行输入	
	48344		输入错误，重新进行输入	
	9581812		输入错误，重新进行输入	
	suqiad		输入错误，重新进行输入	
	wef78		输入错误，重新进行输入	
	e29fq		输入错误，重新进行输入	
	^#@5fe		输入错误，重新进行输入	
	-1		输入错误，重新进行输入	
	-2		输入错误，重新进行输入	
	-3		输入错误，重新进行输入	
	-4		输入错误，重新进行输入	
	-5		输入错误，重新进行输入	

8 系统完善

私人火箭运营模拟系统基本模拟了简化的火箭发射流程，但由于编程环境及文集形式存储记录限制，系统健壮性及运行效率较低，在学习了面向对象程序设

计及数据库原理等计算机相关课程后，可进一步完善系统，达到实用目标。在此讨论系统的如下问题，将考虑如何改进。

8.1 基础数据变动后容易丢失

问题内容：用户在使用发射火箭或自定义修改资金声望的功能时，如果中途退出程序，就会导致资金及声望的数值没有记录进文件中。

问题分析：这是由于资金和声望在运算时是以动态变量的形式参与的，如果不及时将数据写入文件，就会导致程序关闭的时候，数据从内存中丢失

解决方法：在每一次对动态数据操作完毕之后，都立即使用一次 `Savebasicdata` 函数来将当前的动态变量写入文件中，这样就可以避免数据的丢失。

8.2 用户使用时界面跳转过快导致没有浏览完信息

问题内容：用户在使用程序的时候，有部分界面的跳转过快，用户无法及时反应过来程序就已经进行到下一步，导致错过重要信息。

问题分析：这是由于用户在输入值过后函数就立即结束了，从而表现为界面的一闪而过。

解决方法：由于回车键也是一个字符，故在每一个界面显示完毕以后都加入两个 `getchar()`，用户在浏览完毕所有信息后按键盘上的任意一个键位就可进行下一个操作。

8.3 输入非法数据后容易造成死循环

问题内容：在使用程序时，如果用户没有按照界面的输入指引键入了一个非法的数据，就会导致整个程序进入死循环。

问题分析：这是由于键盘上输入非法数据后，键盘当前的输入缓冲区依然存放有非法数据，而导致新的数据无法输入。

解决方法：在每一个 `scanf` 函数之前都使用一次 `fflush` 函数来刷新一次输入缓冲区，就可以让后续的数据输入进来

参考文献

1. 嫦娥 5 号. 百度百科.

<https://baike.baidu.com/item/%E5%AB%A6%E5%A8%A5%E4%BA%94%E5%8F%B7/9026516?fr=aladdin>