青岛理工大学

操作系统课程设计报告

院系：

专业：

学生姓名:

班级：

学号:

题目：

起迄日期:

指导教师:

完成日期: 年 月 日

说明：

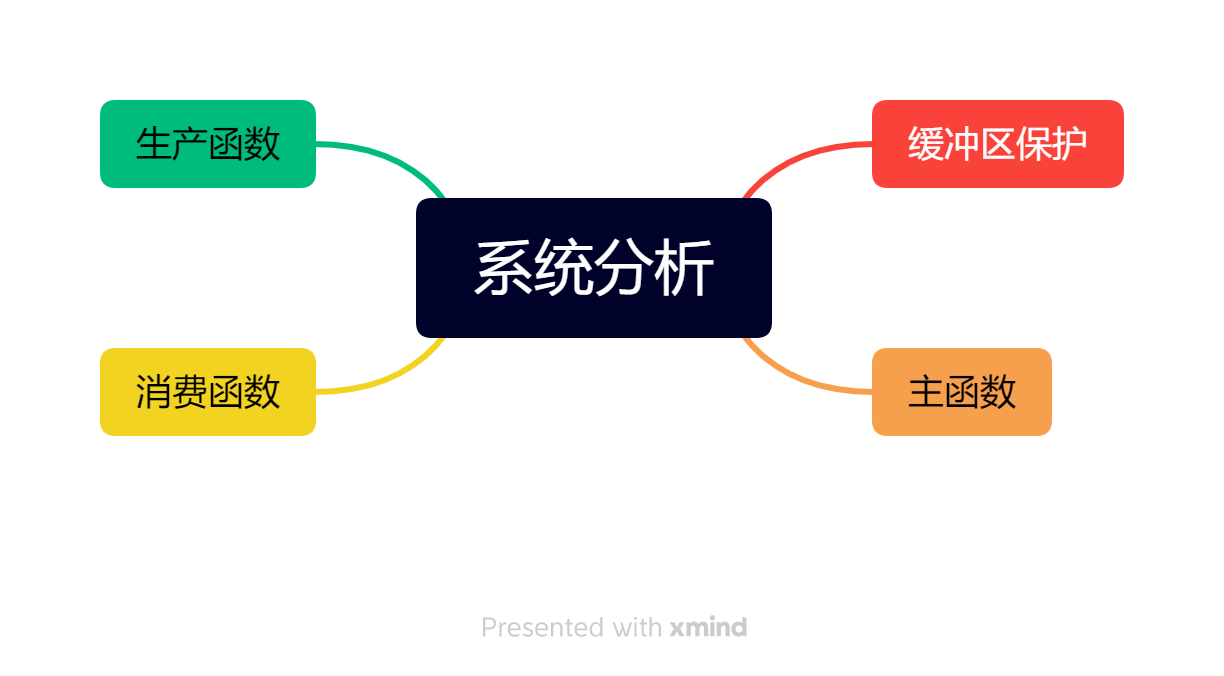
1. 红色字说明，最终报告须删除这部分；
2. 蓝色字的说明，是要用自己所做的具体内容替换的部分。
3. **课程设计目的（标题宋体加粗四号）**

为掌握Linux环境下C语言程序设计， 同时掌握操作系统中信号量互斥与同步的操作，通过在Linux环境下设计生产者消费者问题，达到学习Linux C语言系统编程的目的。

1. **课程设计内容与要求**

**在Linux环境下使用C语言设计生产者消费者程序，实现并发问题的控制。缓冲区有20个单元，每个单元是一个1-20之间的整数，在每个生产者或消费者操作完的同时，输出缓冲区的所有内容。实现多个producer和consumer同时操作。**

1. **系统分析与设计**
2. **系统分析**

****

**系统主要分为四个模块**

1. **生产函数，进行数据的生产，向缓冲区插入数据**
2. **消费函数，进行数据的消费，从缓冲区取出数据**
3. **缓冲区保护，P操作和V操作和一些互斥变量，对缓冲区进行保护**
4. **主函数，驱动生产者和消费者**
5. **系统设计：**
6. **设计整形缓冲区，大小为20，数据为1-20之间的随机数**
7. **设计Macro宏变量进行程序数量的控制，如控制缓冲区大小、控制生产者消费者数量，通过使用宏变量进行控制可以方便程序修改**
8. **设计互斥变量操作保证临界区的数据安全**
9. **设计Consume函数进行数据的消费，使用for循环进行缓冲区的遍历，同时打印出缓冲区所有数据，当前消费数据，当前指针等数据，整个过程都需要在持锁环境下操作。**
10. **设计Produce函数进行数据的生产，使用for循环进行缓冲区的遍历，同时打印出缓冲区所有数据，当前生产数据，当前指针等数据，整个过程都需要在持锁环境下操作。**
11. **设计Main函数，使用for循环和thread操作创建多个生产者和消费者，同时创建互斥变量mutex，P操作和V操作保证临界区的数据安全。**

**使用到的数据结构有，**

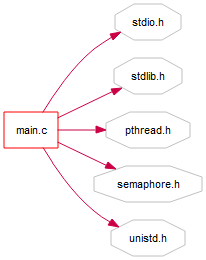
**thread\_t进行线程id的存储，方便后期引用**

**Mutex、sem\_t进行互斥变量和信号量的描述**

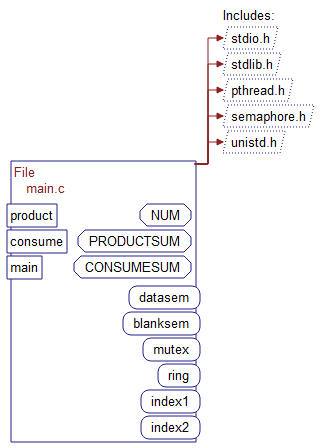
**Int[] 进行缓冲区的表示**

**2.1、模块设计：**

**头文件如下**



宏定义和变量定义如下



**2.2、数据结构说明：**

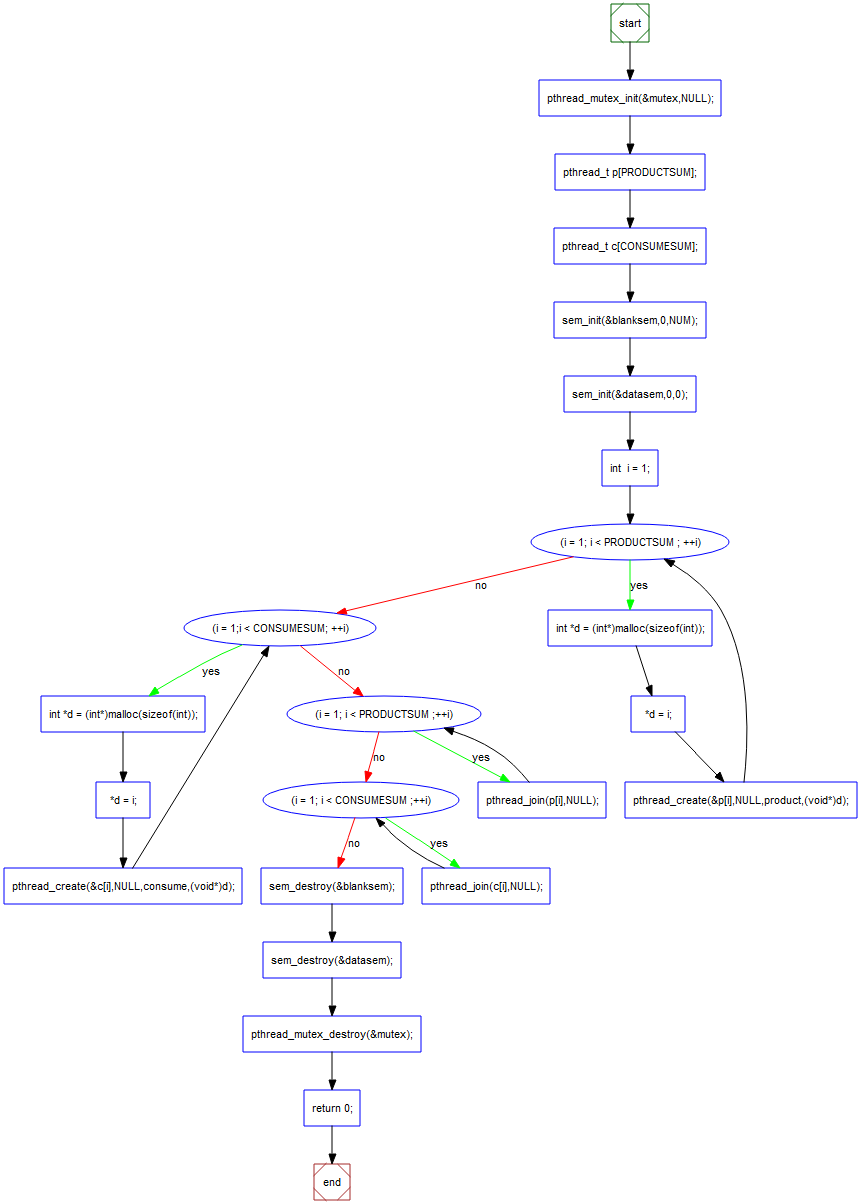
sem\_t datasem;//数据信号量

sem\_t blanksem;//空闲空间信号量

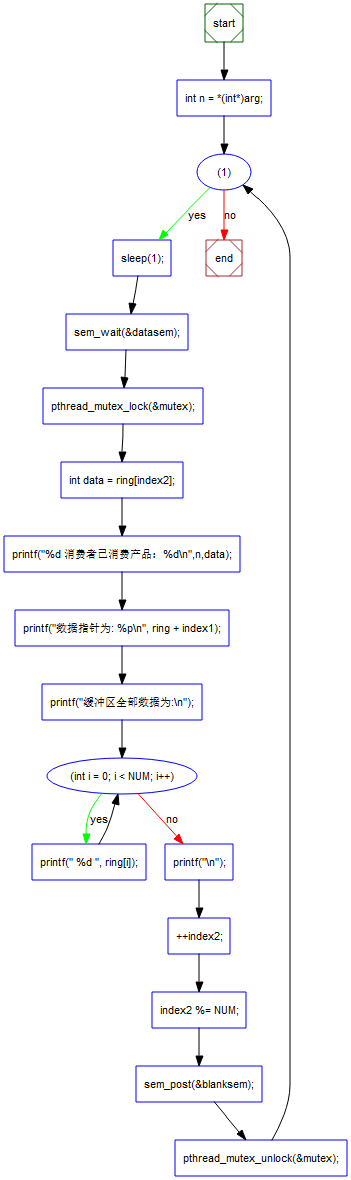
pthread\_mutex\_t mutex;//互斥锁

**2.3、算法流程图：**

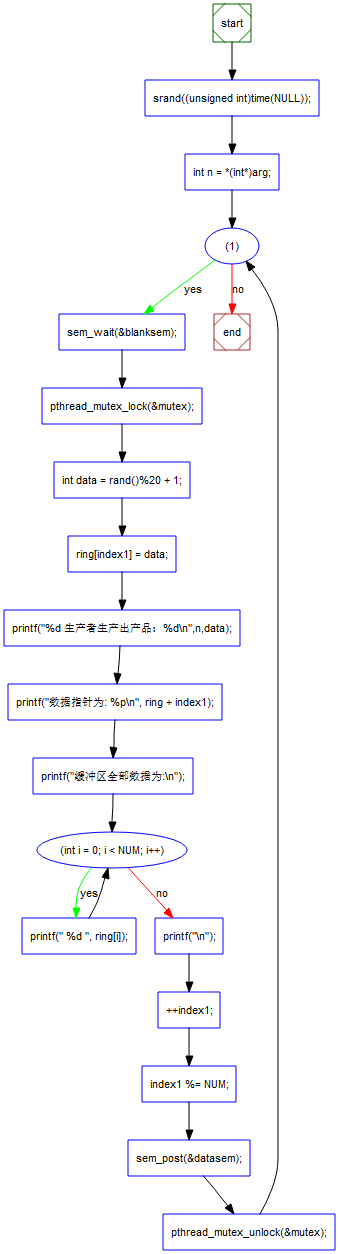
main函数流程图如下，



消费函数流程图如下，



生产函数流程图如下，



**四、系统测试与调试分析**

**1、系统测试**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试说明 | 测试名称 | 生产者消费者 | | |
| 测试目的 | 验证生产者消费者是否正常工作 | | |
| 测试技术 | 单元测试 | | |
| 测试方法 | 黑盒测试法 | | |
| 测试用例 | 测试内容 | 缓冲区数据 | | |
| 测试步骤 | 生产者产出数据 | 消费者消费数据 | 消费者消费数据 |
| 测试数据 | 10 | 20 | 14 |
| 预期结果 | 缓冲区出现10 | 缓冲区中数据20被删除 | 缓冲区数据14被删除 |
| 测试结果 | 与预期相符 | 与预期相符 | 与预期相符 |

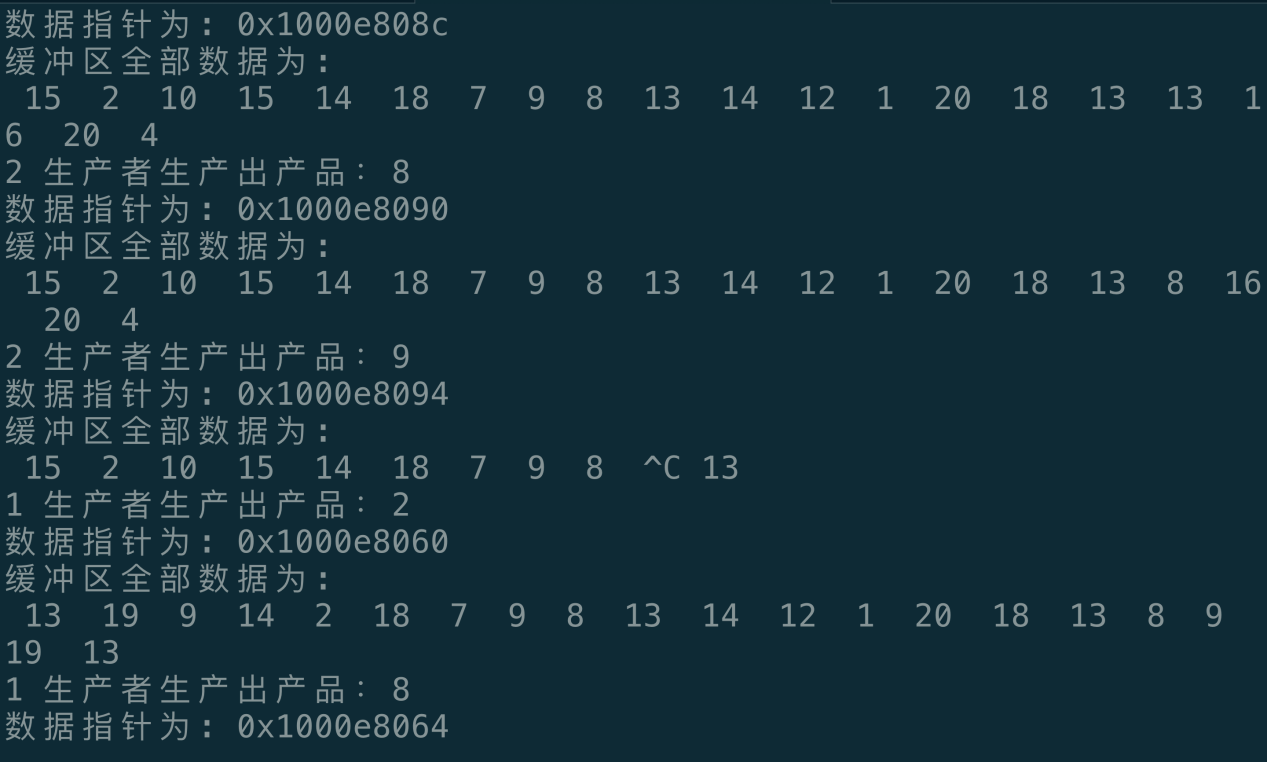
1. **调试分析：**

**调试结果**

在main函数下断点



**运行结果**

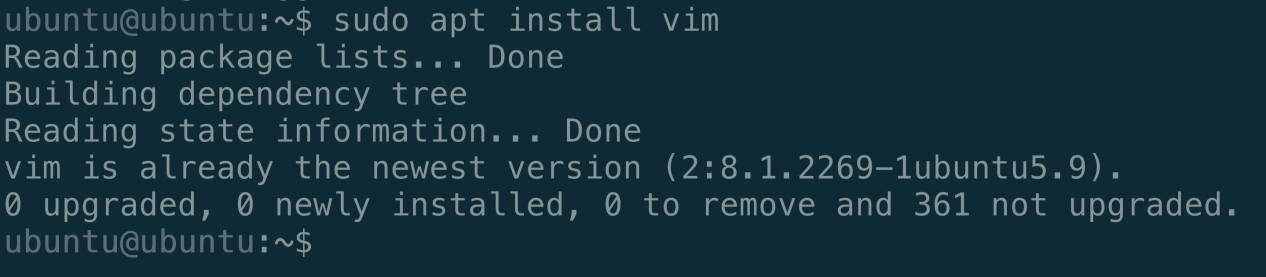


**五、用户手册**

1、安装Linux

2、安装gcc编译器

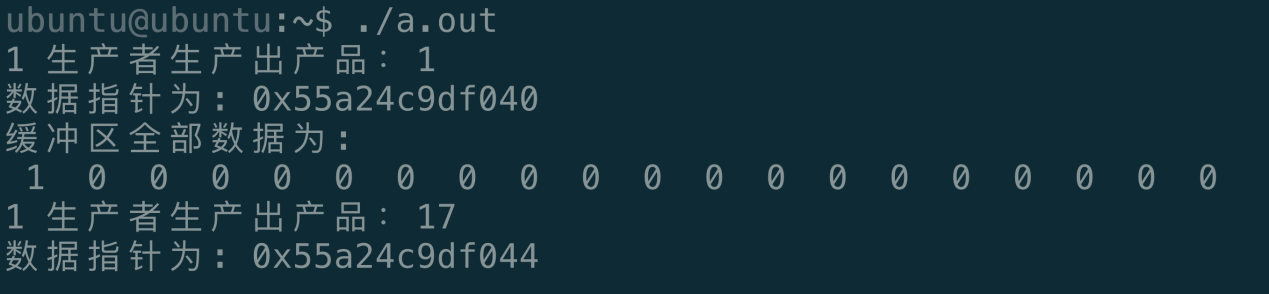
1. 安装vim编辑器



1. 编译程序



1. 运行程序



1. **程序清单**

初始化信号量

sem\_init(&blanksem,0,NUM);//初始化生产者空闲数量为NUM

sem\_init(&datasem,0,0);//消费者为0

创建生产者

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

{

int \*d = (int\*)malloc(sizeof(int));//i是局部变量，如果可能当你将i值传进去时

//时间片切出去将会把i更改，这有就会影响i的值

\*d = i;

pthread\_create(&p[i],NULL,product,(void\*)d);

}

创建消费者

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

{

int \*d = (int\*)malloc(sizeof(int));

\*d = i;

pthread\_create(&c[i],NULL,consume,(void\*)d);

}

互斥变量销毁

sem\_destroy(&blanksem);

sem\_destroy(&datasem);

pthread\_mutex\_destroy(&mutex);

**七、体会与自我评价**

通过本次Linux环境下生产者消费者模型的编程，我的收获非常大。首先在Linux环境下使用C语言进行程序编写就是一件十分不容易的事情。通过使用vmware workstation进行Linux环境的搭建和Linux内部环境的配置，我熟练地掌握了Linux系统以及程序的安装和处理。安装的途中也出了很多差错，原本以为点几下就可以完成的事情，可到时候却可能遇到很多意料之外的事情。首先安装的时候有部分操作系统需要在网络上下载一些软件包，而系统默认是使用的国外的软件源，国内下载的十分的慢，等了几十分钟也只下载了非常少的一部分，最后跳过下载，在系统安装完后修改完软件源后则很快下载好了。

另一个问题是，安装Linux环境下的C语言和C++语言的编译器Gcc的时候提示找不到最基本的头文件，在网络上找了很多方法都没法解决，虽有也用sudo apt purge gcc进行gcc的卸载，然后再用sudo apt install gcc 进行重新安装也无法解决。通过google查询到有人说是因为网络软件包版本的不一致导致了程序冲突，最后只能通过gcc源代码进行编译，编译的过程中已不是一帆风顺的，中途遇到了很多软件包缺失的错误，最后通过apt进行逐一安装，最后完成了Gcc的编译。

从项目中学到了许多关于操作系统多线程编程的知识，同时也更新了我很多的认知。Linux环境下使用C语言创建线程可以用Posix的标准库函数pthread\_create()进行线程的创建，使用pthread\_join()函数进行线程的等待。当我们需要使用大量现成的时候我们可以创建pthread\_t类型数组，然后将每次创建完成的线程ID存放到该数据结构中，方便后面继续访问。

保护临界区的方式也有很多，比如果说互斥体mutex或者信号量semaphore加上PV操作，当然也可以使用其他方式，比如说自旋锁，原子操作等。

**八、参考文献**

　[1]刘燚荣.嵌入式系统高级编程语言技术的实际作用[J].电脑编程技巧与维护,2021(04):28-30.

　　[2]孙丽娟,唐伟强,黄小丽,龙文堃.嵌入式系统在起重机吊具防摇控制中的应用[J].兰州理工大学学报,2022,46(02):97-102

　　[3]杨凯.嵌入式系统自动化测试框架的设计与实现[J].福建电脑,2022,36(03):60-63.

　　[4]柏文杰,赵岗岗,张圣道,陈晨.Linux系统的多层次菜单设计[J].工业控制计算机,2021,33(03):52+54.

　　[5]李鼎,张秀芳,刘洲洲.基于Nucleus的嵌入式系统数据库架构设计[J].微型电脑应用,2021,36(03):28-30.

**九、课程设计评价（由任课教师填写）**

|  |  |
| --- | --- |
| 课程设计评价 | 成绩：教师：  年 月 日 |