操作系统课程设计

**超市购物模拟**

**组别： 5**

**组员：学号 1613042019 姓名 丁建宇 编程**

**学号 1613042023 姓名 江彦君 编程**

**学号 1613042022 姓名 范益斌 设计报告**

**小组共同完成汇报 ,PPT**

**南通大学·计算机科学与技术学院**

**2018年06月**

**摘要**

**本设计采用了****visual studio 2015开发工具，解决了超市购物模拟问题，实现了顾客随机购物、顾客结账、超市抽奖、显示收据、取还购物车等功能。经分析表明，程序逻辑正确。程序具有。。。。的特点。**

**关键词：超市购物模拟、Windows 线程创建、控制、同步与互斥、信号量**

**排名：**

目录

[**1.** **实验题目：超市购物模拟** 1](#_Toc518502519)

[**2.** **实验工具** 1](#_Toc518502520)

[**3.** **问题描述、分析和功能设计** 1](#_Toc518502521)

[3.1 问题描述 1](#_Toc518502522)

[3.2 分析及功能设计 1](#_Toc518502523)

[1） 问题分析 1](#_Toc518502524)

[2） 功能设计 1](#_Toc518502525)

[**4.** **测试用例及其求解** 2](#_Toc518502526)

[4.1 测试数据 2](#_Toc518502527)

[4.2 操作分析 2](#_Toc518502528)

[**5.** **程序设计** 2](#_Toc518502529)

[5.1 逻辑结构设计 2](#_Toc518502530)

[5.2 存储结构设计 4](#_Toc518502531)

[5.3 算法设计 4](#_Toc518502532)

[**6.** **程序运行情况** 5](#_Toc518502533)

[**7.** **总结** 5](#_Toc518502534)

[**参考文献：** 5](#_Toc518502535)

1. **实验题目：超市购物模拟**
2. **实验工具**

编程环境：visual studio 2015

编程语言：visual C++

设计报告：OFFICE 2016之word

汇报方档：OFFICE 2016之 PowerPoint

1. **问题描述、分析和功能设计**
   1. 问题描述

编写一个程序，用线程控制和线程同步（使用互斥体和信号量）机制来模拟超市购物的运转情况。

* 1. 分析及功能设计

1. 问题分析

对于超市购物来说，可以从超市管理员和顾客两方面入手。

超市管理员负责：

* 开放超市;
* 创建各个顾客线程;
* 关闭超市;
* 在结束模拟之前，等待商店中所有顾客付钱离开。

顾客的行动有：

* 等待进人超市;
* 进行数量随机的商品选购;
* 在结账柜台排队等待付款;
* 在结账柜台处花费是随机的;
* 离开结账柜台;
* 离开超市。

1. 功能设计
2. 创建一个代表超市管理员的线程。
3. 创建顾客线程后开始购物。
4. 允许若干顾客线程并发购物。
5. 在模拟中,熟食柜台中只有一名店员,所有顾客对于熟食柜台的同步是用互斥体来保证。
6. 模拟中，结账柜台的个数是固定的，用信号量管理。

此外，我们组对超市购物模拟进行了改进和优化。

* 1. 若顾客购买商品预付金额大于300，可以得到抽奖的机会。
  2. 在结账完成后，每个顾客可以得到收据。
  3. 顾客可以选择申请购物车购物，申请不到则等待。离开商店后归还购物车。利用信号量管理。

1. **测试用例及其求解**
   1. 测试数据

const int MAXCUS= 20; //最多创建的顾客数目

const int CUSINMAR=15; //最多进人超市的顾客数目

const int MAXQUEUE=5; //各种队列的最大长度(结账队列，买肉队列,进人超市队列)

const int GOODS=5; //随机购物时的货物种类数

const int MAXCOUNTER= 3; //结账柜台的个数

const int cargo=5; //购物车的个数

const int PRICE [GOODS+1]= {5, 100，12,30,500,7};

//0项代表熟食的价格,1~5项代表其他五种商品的价格

const int MAXQUANTITY [GOODS+1]= ｛5,2,10,5,1,7｝;

//0项代表熟食的数量，1~5项代表其他五种商品的数量

* 1. 操作分析

根据以上测试用例，进行对预期结果的分析。

1. **程序设计**
   1. 逻辑结构设计

在超市购物模拟中，Mythread类作为线程类、顾客类和超市管理员类的父类，定义如下：

class MyThread {

protected:

HANDLE myhandle; //句柄

string myname;

static DWORD WINAPI ThreadProc(LPVOID lpParam); //指定线程起始地址

public:

MyThread(string name = "");

virtual ~MyThread();

void start(); //建立一个新线程

virtual void run() = 0;

virtual void WaitForCompletion();

int get\_random();

};

超市类定义如下：

class SuperMarket :public MyThread

{

public:

void output(string os); //在文档和屏幕上显示字符串

SuperMarket(); //构造函数

virtual void run(); //超市线程主体

~SuperMarket(); //析构函数，输出超市线程关闭信号

};

顾客类定义如下：

class Customer :public MyThread

{

protected:

//ThreadNumMutex:信号量,控制进人超市的顾客最多为CUSINMAR

//CounterMutex[] :互斥体，控制互斥地到MAXCOUNTER个柜台结账

//AccessCounter :互斥体，控制互斥的访问counter[]

//CookMutex:互斥体,控制互斥的到熟食队列买熟食

//AccessCookCounter:互斥体，控制互斥的访问cookcounter

//VesselCounter：信号量，控制购物车数量最多为cargo

static HANDLE ThreadNumMutex, CounterMutex[MAXCOUNTER], AccessCounter, CookMutex, AccessCookCounter, VesselCounter;

static int counter[MAXCOUNTER]; //counter[]:统计到各个结账柜台排队的顾客数

int countertime; //每个顾客的结账时间

int receipt[6] = { 0 }; //每个顾客的收据数组

static int cookcounter; //cookcounter:统计到熟食队列排队的顾客数目

int cost; //顾客消费的总额

float discount; //抽奖得到的折扣

bool vessel; //是否需要购物车

void static setCustomerNumber(); //初始化ThreadNumMutex

void static setCounter(); //初始化CounterMutex[]、AccessCounter

void static setCookMutex(); //初始化CookMutex、AccessCookCounter

void static setVessel(); //初始化

void handlecounter(); //处理顾客结账排队的函数

void shopping(); //处理顾客随机购物的函数

public:

Customer(int ID);

void static initialize(); //总初始化操作

virtual void run(); //顾客线程主体

static void closehandle(); //关闭信号量

* 1. 存储结构设计

用vector可变长数组存储创建的各个顾客线程

vector<Customer>cus;

Customer::initialize();

for (int i = 0; i<MAXCUS; i++)

{

cus.push\_back(Customer(i));

}

用一般数组存放商品价格，商品数量。此外，每个顾客有一个receipt数组存放选购各商品的数量。

const int PRICE[GOODS + 1] = { 5,100,12,30,500,7 };

const int MAXQUANTITY[GOODS + 1] = { 5,2,10,5,1,7 };

int receipt[6] = { 0 };

* 1. 算法设计

1. 超市购物抽奖算法

* 算法功能：给部分顾客打折优惠。
* 算法描述：顾客结账前，若cost大于300，则从7、8、9中随机抽取一个数，分别对应着7折，8折，9折的优惠，从而改变cost值；若cost小于等于300，不作为。
* 算法步骤：

Step1：判断cost是否大于300。

Step2：若大于300，利用random（7，9）从中取数；若不大于300，不作为。

Step3：若在random（7，9）中取到数，cost值根据取到的数作出相应改变；

否则cost值不变。

1. 显示顾客收据算法

* 算法功能：显示顾客购买各种商品数
* 算法描述：每个顾客有一个receipt数组存放选购各商品的数量，在结账完成后利用receipt数组显示顾客购买各个商品的数量。
* 算法步骤：

Step1：在顾客随机购物时，将购买的各个商品数量放入receipt数组的对应

位置。

Step2：结账完成后利用receipt数组显示顾客购买各个商品的数量。

1. 申请/归还购物车算法

* 算法功能：为顾客提供取还购物车服务。利用信号量管理。
* 算法描述：用布尔型vessel标记顾客是否需要购物车，顾客线程创建时，若vessel为0，执行shopping（）和handlecounter（），即正常购物动作；若vessel为1， 则增加一个申请使用购物车的步骤，成功取得购物车，则购物车数减一，若购物车数为0时，顾客等待，直到获得购物车再进入超市购物，顾客离开超时后归还购物车。
* 算法步骤：

Step1：设置布尔型vessel判断顾客是否需要购物车。

Step2：若vessel==1，则需要申请一个购物车；若vessel==0，则不需要购物

车，直接进入超市正常购物。

Step3：申请购物车后，若顾客成功得到购物车，则cargo--，进入超市购物；

若cargo==0，申请不成功，顾客等待，直到获得购物车再进入超市购物。顾

客离开超时后归还购物车，cargo++。

1. **程序运行情况**
2. **总结**

**参考文献：**

1. 百度知道.createthread()传递参数问题.

https://baike.baidu.com/item/CreateThread/8222652,2018.4.12

1. CSDN博客.虚函数.

https://blog.csdn.net/zhuoya\_/article/details/80294956,2018.5.13