

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

**МНОГОПОТОЧНОЕ КОНСОЛЬНОЕ ПРИЛОЖЕНИЕ, МОДЕЛИРУЮЩЕЕ  
РАБОЧИЙ ДЕНЬ СУПЕРМАРКЕТА**

Пояснительная записка

**Исполнитель:**

Студентка группы БПИ195

\_\_\_\_\_/Зубарева Н.Д./

«13» декабря 2020 г.

## Оглавление

1. Текст задания.....	2
2. Применяемые расчетные методы .....	3
2.1. Теория решения задания .....	3
2.2. Организация многопоточности.....	3
2.3. Ввод входных данных .....	3
2.4. Вывод данных.....	3
3. Тестирование программы.....	4
3.1. Корректные значения .....	4
3.2. Некорректные значения .....	5
4. Список литературы.....	6

# 1. Текст задания

Вариант 10: Задача о супермаркете. В супермаркете работают два кассира, покупатели заходят в супермаркет, делают покупки и становятся в очередь к случайному кассиру. Пока очередь пуста, кассир спит, как только появляется покупатель, кассир просыпается. Покупатель спит в очереди, пока не подойдет к кассиру. Создать многопоточное приложение, моделирующее рабочий день супермаркета.

## 2. Применяемые расчетные методы

### 2.1. Теория решения задания

По условию требуется создавать потоки для работы покупателей и кассиров в магазине. В этом приложении это реализовано с помощью `std::thread` [3], мьютексов и семафоров.

### 2.2. Организация многопоточности

Программа реализует модель параллелизма клиент-сервер [2]: продавцы – это класс, содержащий внутри себя поток, осуществляющий работу с очередью экземпляров класса покупателей, то есть работающий с покупателями, пока очередь не пуста, спящий иначе. Класс покупателя содержит поток, с помощью которого покупатель встает в очередь, ждет, пока не дойдет до продавца (спит) и затем уходит. И у продавца, и у покупателя есть семафоры, по которым осуществляется уведомление их о чем-то, что происходит – например уведомление покупателя о том, что пришла его очередь в кассе. У продавца так же есть мьютекс на очередь к его кассе, чтобы ее изменение происходило корректно. Также в программе есть поток – будильник, который реализует длину рабочего дня в магазине, когда он срабатывает, продавцы уходят по домам.

Чтобы было интереснее посмотреть на работу продавца с очередью, покупатели приходят и встают в очередь в три волны: до начала работы потоков продавцов, во время работы и незадолго до конца, поэтому обработаны ситуации появления новых покупателей после того, как изначальная очередь опустела и продавец заснул, также случай, когда покупатель приходит после окончания работы продавца.

### 2.3. Ввод входных данных

Ввод данных осуществлен через командную строку, хотя это реализация дополнительного функционала и в принципе это можно было не делать. В командной строке задается число продавцов, покупателей и длительность рабочего дня в секундах. Если аргументы не заданы, то при симуляции они будут равны дефолтным значениям 2, 30, 10 соответственно. Есть проверки на корректность вводимых чисел (они все преобразуются из строк через `std::stoi`), при возникновении исключений задаются дефолтные значения, нижние границы для параметров определены как 1, 1, 3, максимальные – как 10, 100, 30, чтобы программа не была слишком долгой.

### 2.4. Вывод данных

Выходные данные представлены логами в консоли: каждый поток, будь он клиентом или покупателем, выводит строковую информацию о том, что он делает как свой идентификационный номер и действие, например «Cashier 1 is working with customer 1...» - кассир 1 обслуживает 1 покупателя, или «Customer 10 is going home having bought their groceries» - покупателя 10 обслужили, и он уходит из магазина.

### 3. Тестирование программы

Программа компилируется и запускается следующим образом из командной строки (рисунок 1).

```
nat@LAPTOP-1AGP7LH0: /mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2
nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ g++ -pthread main.cpp
nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out
-----WE ARE SIMULATING A SHOP WITH 2 CASHIERS, 30 CUSTOMERS AND 10 SEC WORKDAY DURATION-----
```

Рисунок 1 Команды компиляции и запуска программы

#### 3.1. Корректные значения

Программа осуществляет симуляцию работы магазина, выводя логи того, что происходило с продавцами и покупателями. Аргументы командной строки можно не задавать (рисунок 2), тогда значения будут дефолтными.

```
nat@LAPTOP-1AGP7LH0: /mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2
nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out
-----WE ARE SIMULATING A SHOP WITH 2 CASHIERS, 30 CUSTOMERS AND 10 SEC WORKDAY DURATION-----

-----
THE FIRST WAVE OF CUSTOMERS ARRIVES EARLY
-----

Customer 1 arrives!
Customer 1 is waiting in line and sleeping

Customer 2 arrives!
Customer 2 is waiting in line and sleeping

Customer 3 arrives!
Customer 3 is waiting in line and sleeping

Customer 4 arrives!
Customer 4 is waiting in line and sleeping

Customer 5 arrives!
Customer 5 is waiting in line and sleeping

Customer 6 arrives!
Customer 6 is waiting in line and sleeping

Customer 7 arrives!
Customer 7 is waiting in line and sleeping

Customer 8 arrives!
Customer 8 is waiting in line and sleeping

Customer 9 arrives!
Customer 9 is waiting in line and sleeping

Cashier 1 starts their workday!

Customer 10 arrives!
Customer 10 is waiting in line and sleeping

Cashier 2 starts their workday!

Cashier 1 is working with customer 1...
```

Рисунок 2 Начало работы программы при дефолтных параметрах

Если задать корректные аргументы, программа будет работать с ними (рисунок 3, рисунок 4).

```
nat@LAPTOP-1AGP7LH0: /mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out 2 2 10
-----WE ARE SIMULATING A SHOP WITH 2 CASHIERS, 2 CUSTOMERS AND 10 SEC WORKDAY DURATION-----

-----
THE FIRST WAVE OF CUSTOMERS ARRIVES EARLY
-----

Cashier 1 starts their workday!
Cashier 2 starts their workday!

The 1 line is empty, so cashier 1 is sleeping

-----
THE SECOND WAVE OF CUSTOMERS ARRIVES AT NORMAL TIME
-----

The 2 line is empty, so cashier 2 is sleeping

Customer 1 arrives!
Customer 1 is waiting in line and sleeping

Cashier 1 is working with customer 1...
```

Рисунок 3 Работа программы при корректных данных, начало

```

Cashier 1 is working with customer 1...
Customer 1 is going home having bought their groceries
The 1 line is empty, so cashier 1 is sleeping

-----
THE THIRD WAVE OF CUSTOMERS ARRIVES ALMOST LATE
-----

Customer 2 arrives!
Customer 2 is waiting in line and sleeping

Cashier 2 is working with customer 2...
Customer 2 is going home having bought their groceries
The 2 line is empty, so cashier 2 is sleeping

THE WORKDAY IS OVER!!!

Cashier 1 is going home
Cashier 2 is going home
nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$

```

Рисунок 4 Работа программы при корректных данных, окончание

### 3.2. Некорректные значения

В базовой реализации все всегда работает корректно, при задании аргументов командной строки – количества покупателей, продавцов и времени работы магазина могут происходить ошибки, обрабатываемые заданием дефолтных или максимальных\минимальных значений параметров вместо неверных (рисунок 5, рисунок 6, рисунок 7).

```

nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out 0 0 0
0 cashiers is a wrong thing, we'll run the simulation with 2
0 customers is a wrong thing, we'll run the simulation with 30
0 workday duration is a wrong thing, we'll run the simulation with 10 seconds
-----WE ARE SIMULATING A SHOP WITH 2 CASHIERS, 30 CUSTOMERS AND 10 SEC WORKDAY DURATION-----

-----
THE FIRST WAVE OF CUSTOMERS ARRIVES EARLY
-----

```

Рисунок 5 Работа программы при нечисловых параметрах

```

nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out 200 200 100
200 cashiers is a lot, we'll run the simulation with 10
200 customers is a lot, we'll run the simulation with 100
100 seconds is a lot, we'll run the simulation with 30 seconds
-----WE ARE SIMULATING A SHOP WITH 10 CASHIERS, 100 CUSTOMERS AND 30 SEC WORKDAY DURATION-----

-----
THE FIRST WAVE OF CUSTOMERS ARRIVES EARLY
-----

```

Рисунок 6 Работа программы при параметрах больше дозволенных

```

nat@LAPTOP-1AGP7LH0:/mnt/c/Users/Natalya/Desktop/TheCodeSoFar/HSE/assembler/miniproject2$ ./a.out -2 -2 -10
-2 cashiers is too little, we'll run the simulation with 1
-2 customers is too little, we'll run the simulation with 1
-10 seconds is too little, we'll run the simulation with 3 seconds
-----WE ARE SIMULATING A SHOP WITH 1 CASHIERS, 1 CUSTOMERS AND 3 SEC WORKDAY DURATION-----

-----
THE FIRST WAVE OF CUSTOMERS ARRIVES EARLY
-----

```

Рисунок 7 Работа программы при параметрах меньше дозволенных

## 4. Список литературы

- [1] Инструкция по составлению пояснительной записки [Электронный ресурс]. //URL: <http://softcraft.ru/edu/comparch/tasks/mp01/> (Дата обращения: 12.12.2020, режим доступа: свободный)
- [2] Практические приемы построения многопоточных приложений [Электронный ресурс]. //URL: <http://softcraft.ru/edu/comparch/tasks/t03/> (Дата обращения: 12.12.2020, режим доступа: свободный)
- [3] Std::thread::thread [Электронный ресурс]. //URL: <https://en.cppreference.com/w/cpp/thread/thread/thread> (Дата обращения: 12.12.2020, режим доступа: свободный)