Пояснительная записка (Д/З №5)

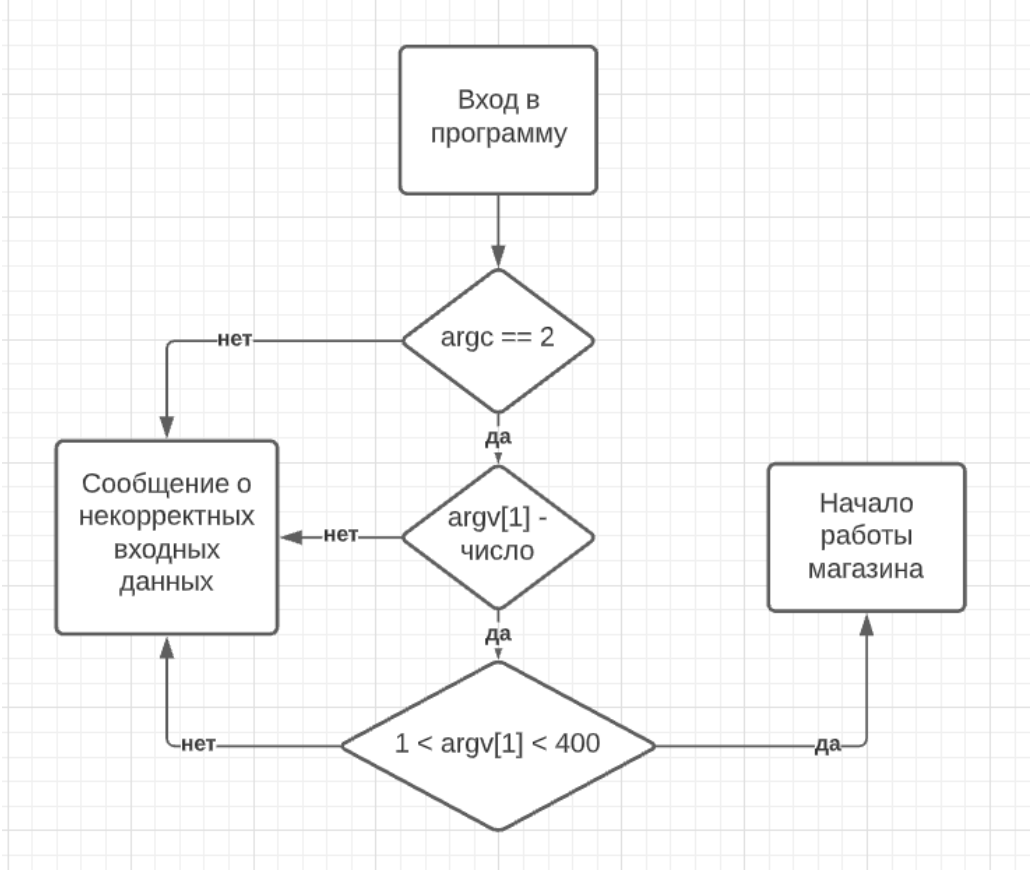
Глазков Максим БПИ208 - Вариант 9

1. Описание задания

Задача о магазине - 1. В магазине работают три отдела, каждый отдел обслуживает один продавец. Покупатель, зайдя в магазин, делает покупки в произвольных отделах, и, если в выбранном отделе продавец не свободен, покупатель становится в очередь и засыпает, пока продавец не освободится. Создать многопоточное приложение, моделирующее рабочий день магазина.

2. Описание процесса работы программы

При старте программы происходит проверка входных данных (выполняется в основном потоке)



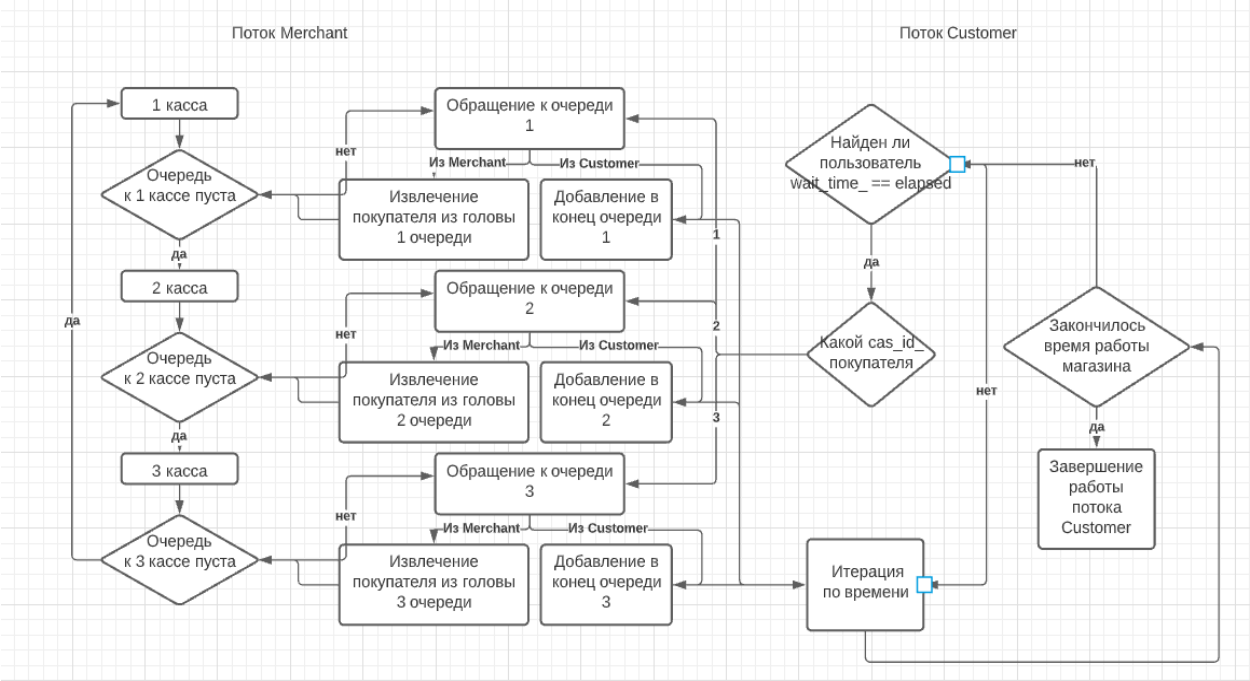
После проверки входных данных в основном потоке создаются следующие сущности:

- Мьютекс
- Объект класса Merchant олицетворяющий продавца магазина (в нем хранится только поле cash_id_, показывающей с какой кассой сейчас работает торговец)
- Список покупателей vector<Customers>. Customer – объект олицетворяющий одного покупателя (в нем хранятся wait_time_ - количество времени, по прошествии которого покупатель войдет в магазин, id_ - уникальный номер покупателя, cash_id_ - номер кассы, к которой подойдет покупатель)

- Queue1, queue2, queue3 (queue<Customer>) – очереди покупателей к первой, второй и третьей кассе соответственно

Затем в программа разделяется на два потока Merchant (merch) и Customer (customer_process). Схематическое представление работы потоков:

(elapsed – время, которое программа проработала (или проработал магазин))



Поток Merchant, как и поток Customer, заканчивает свою работу по окончанию времени работы магазина (заданное время работы магазина 500 секунд, но оно не детерминировано -> зависит от наплыва клиентов)

После окончания работы потоков Merchant и Customer заканчивается процесс работы программы

3. Входные и выходные данные

На ввод подается консольная команда следующего вида: <Программа.exe> <N>, где N – целое положительное число в промежутке от 1 до 400 включительно

На вывод программа может выдавать следующие строки в консоли:

- “Preparing shop…” – выводиться в начале работы программы
- “[Merchant thread]: Merchant working with <i> cashbox” – строка показывает, что сейчас продавец находиться на i-ой кассе (i – число от 1 до 3) -> через 1 секунду переход на следующую кассу
- “[Customer thread]: Customer <id> arrived to cashbox <i>” – строка показывает, что покупатель с определенным id встал в очередь к i-ой кассе
- “[Merchant thread]: Merchant works on <i> cashbox with <id> customer...” – демонстрирует, что сейчас продавец работает с покупателем с определенным id на i-ой кассе -> работа с покупателем от 2 до 5 секунд
- “[Merchant thread]: Customer <id> was served successfully” – показывает, что работа с покупателем с id закончена (всегда выводиться после предыдущей строки в потоке

Merchant, естественно, что во время работы с покупателем могут прийти другие покупатели)

В начале и при окончании работы программы покупатели не будут приходить (это время подготовки магазина к рабочему дня и окончания работы магазина после рабочего дня)

4. Дополнительные сведения

Вывод программы может быть искажен (обычно один раз за отработку программы), так как потоки Customer и Merchant работают независимо и по моей логике не должны иметь в этом плане никаких ограничений (покупатель же может прийти в тот момент, когда продавец только перебежал на другую кассу или только начал работу с другим покупателем)

Есть мьютекс на добавление элемента в очередь (это единственное ограничение, чтобы продавец не пропускал покупателей, которые только добавились в очередь)

Программа может работать 500–600 секунд