

Глебов Павел Алексеевич. БПИ228. Вариант 22. 7 баллов.

Условие задачи:

Задача о картинной галерее. Вахтер следит за тем, чтобы в картинной галерее одновременно было не более 25 посетителей. Для обозрения представлены 5 картин. Каждый посетитель случайно переходит от картины к картине, но, если на желаемую картину любуются более пяти посетителей, он стоит в стороне и ждет, пока число желающих увидеть эту картину не станет меньше. Посетитель покидает галерею по завершении осмотра всех картин. Каждый посетитель уникальный, то есть имеет свой номер (например, PID). В галерею также пытаются постоянно зайти новые посетители, которые ожидают своей очереди и разрешения от вахтера, если та заполнена.

Создать клиент–серверное приложение, моделирующее однодневную работу картинной галереи (например, можно ограничить числом посетителей от 100 до 300). Вахтер — сервер. Картинная галерея — клиент. Приходящие посетители реализуются общим клиентом. Время нахождения возле картины для каждого посетителя является случайной величиной в некотором диапазоне.

Описание на 4-5 баллов:

Исходные сущности:

1. Сервер (Server):

- Создает и управляет галереей.
- Принимает входящие соединения от посетителей.
- Использует мьютексы для синхронизации доступа к общим ресурсам (общее количество посетителей, количество просматривающих картину).

2. Посетители (Visitor):

- Каждый посетитель представляет собой отдельный процесс или поток.
- Посетитель подключается к серверу, отправляет команды на вход в галерею (ENTER), просмотр картин (MOVE), и выход из галереи (LEAVE).
- Используются сокеты для установки соединения с сервером.

Взаимодействие между сущностями:

1. Запуск сервера:

- Сервер запускается с указанием IP-адреса и порта.
- Создается сокет и выполняется привязка к заданному IP-адресу и порту.
- Сервер прослушивает входящие соединения и создает новый поток или процесс для каждого подключившегося посетителя.

2. Запуск посетителей:

- Каждый посетитель запускается как отдельный поток.
- Посетители соединяются с сервером по IP-адресу и порту.
- Отправляют команды серверу: ENTER для входа в галерею, MOVE для просмотра конкретной картине, LEAVE для выхода из галереи.
- В случае, если в галерее нет места (максимальное количество посетителей превышено), посетитель получает ответ WAIT и ожидает до тех пор, пока место не освободится.

3. Серверная логика:

- При получении команды ENTER сервер проверяет, есть ли место в галерее и разрешает вход либо указывает посетителю подождать.
- При команде MOVE сервер проверяет, доступно ли место для просмотра картины и разрешает доступ либо указывает посетителю подождать.
- Когда посетитель завершает просмотр картин, он отправляет команду LEAVE и освобождает место для других посетителей.

4. Синхронизация и безопасность:

- Используется мьютекс для защиты общих данных (количество посетителей в галерее, количество просматривающих каждую картину) от одновременного доступа из разных потоков или процессов.
- Каждый сокет обрабатывается в отдельном потоке, что позволяет одновременно обслуживать несколько посетителей.

Ход выполнения:

1. Запуск сервера:

- Сервер создает сокет, привязывается к порту и начинает прослушивать входящие соединения.
- Для каждого входящего соединения создается новый поток, который обрабатывает команды посетителя.

2. Запуск посетителей:

- Каждый посетитель отправляет команды на вход, просмотр картин и выход из галереи.
- Посетители случайным образом выбирают картину для просмотра и могут ждать, если картину уже просматривают максимально допустимое количество посетителей.

3. Завершение программы:

- После того как все посетители завершат просмотр и выйдут из галереи, программа завершает выполнение.

Описание на 6-7 баллов:

1. Запуск сервера (infoServer):

- Сервер запускается и начинает прослушивать входящие соединения.
- При подключении информационного клиента сервер выводит сообщение "Information client connected."

2. Запуск посетителей (visitor):

- Каждый посетитель подключается к серверу и отправляет команды.
- В случае успеха или отказа в команде сервер отправляет соответствующее сообщение информационному клиенту.

3. Работа информационного клиента (infoVisitor):

- Информационный клиент подключается к серверу и получает все данные о действиях посетителей.

- Информация отображается в реальном времени, что позволяет отслеживать состояние системы.