

# 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:

- **Основной сервер (server.c):**
  - Запускается на определенном порту.
  - Создает UDP-сокеты для приема запросов от посетителей.
  - Использует мьютекс для синхронизации доступа к общим данным (текущее количество посетителей и количество зрителей у каждой картины).
  - После своего запуска ожидает запросы от клиентов.

#### 2. Клиенты:

- **Посетители (visitor.c):**
  - Каждый посетитель работает в отдельном потоке.
  - Создаются и отправляются UDP-запросы на сервер, используя команды ENTER, MOVE, LEAVE.
  - Если сервер отвечает WAIT, клиент ждет некоторое случайное время перед повторной попыткой.

### Взаимодействие между сущностями:

- **Сервер:**
  - Запускается на указанном порту и начинает принимать запросы от посетителей.
  - Взаимодействует с клиентами, обрабатывая запросы на вход в галерею и просмотр картин.
  - Использует мьютекс для предотвращения одновременного доступа к общим данным.
  - Ведет лог событий о действиях посетителей.
- **Клиенты:**
  - Запускаются в отдельных потоках.

- Имитируют действия посетителей, отправляя UDP-запросы на сервер для входа в галерею, просмотра картин и выхода.
- После отправки каждого запроса ждут ответа от сервера и обрабатывают его в зависимости от содержания ответа.
- Случайно задерживают свои действия, чтобы имитировать разное время, проведенное на просмотре картин.

### **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.