

Домашнее задание №5, вариант 11

Колтунов Кирилл, БПИ207

1. Описание задания:

В гостинице 30 номеров, клиенты гостиницы снимают номер на одну ночь, если в гостинице нет свободных номеров, клиенты устраиваются на ночлег рядом с гостиницей и ждут, пока любой номер не освободится. Создать многопоточное приложение, моделирующее работу гостиницы.

В качестве входного параметра пользователю необходимо в командной строке указать одно значение – количество посетителей гостиницы. Данное значение должно быть больше 10 и не больше 100. При некорректном вводе аргументов командной строки программа выведет сообщение об ошибке.

Далее в процессе работы программы будут выводиться сообщения о происходящих процессах. Либо сообщение о том, что человек заселен в номер, либо о том, что он выселен.

2. Описание используемой модели

Для решения задачи была выбрана модель “Производитель – потребитель”.

Ее суть состоит в том, что процесс-производитель выполняет вычисления и выводит поток результатов. Процесс-потребитель вводит и анализирует поток значений. Многие программы в той или иной форме являются производителями и/или потребителями. Сочетание становится особенно интересным, если производители и потребители объединены в конвейер — последовательность процессов, в которой каждый из них потребляет данные выхода предшественника и производит данные для последующего процесса. Классическим примером являются конвейеры в операционной системе Unix.

Процесс-производитель ожидает (при необходимости), пока в буфере появится свободное место, затем добавляет в конец буфера новую строку. Процесс-потребитель ожидает (при необходимости), пока в буфере не появится строка данных, затем забирает ее.

Для контроля доступа к общему ресурсу были использованы мьютексы и семафоры.

Мьютекс — это базовый механизм синхронизации. Он предназначен для организации взаимоисключающего доступа к общим данным для нескольких потоков с использованием барьеров памяти (для простоты можно считать мьютекс дверью, ведущей к общим данным). То есть, мьютекс служит для того, чтобы в один момент времени в конкретный блок кода мог зайти только один поток.

Семафор — это объект, который используется для контроля доступа нескольких потоков до общего ресурса. В общем случае это какая-то переменная, состояние которой изменяется каждым из потоков. Текущее состояние переменной

определяет доступ к ресурсам. В данной программе присутствуют семафоры `full` – количество занятых ячеек и `empty` – количество свободных ячеек, поток-производитель увеличивает на 1 значение `full` и уменьшает на 1 значение `empty`, поток-потребитель, наоборот, увеличивает `empty` и уменьшает `full`. Когда значения данных семафоров достигают предельных допустимых значений, то при `full = 30`, `empty = 0` блокируются все потоки-производители (посетители), при `full = 0`, `empty = 30` блокируются все потоки-потребители (комнаты). Потоки разблокируются тогда, когда вызовется `sem_post` и значения семафоров перестанут быть равными предельным значениям.

3. Общий принцип работы приложения

Сначала в цикле создаются и запускаются все потоки-посетители, а затем все потоки-комнаты. У каждого потока вызывается его стартовый метод, в котором содержится блокировка (мьютексы). Сначала посетители в произвольном порядке заполняют все комнаты отеля (сначала кого-то селят в первую комнату, потом во вторую и так далее в каждую комнату). После поселения все посетители спят на протяжении рандомного количества секунд от 5 до 10, при этом семафоры блокируют доступ к гостинице другим посетителям, в какой-то момент времени начинают работать потоки-комнаты, которые начинают выселять посетителей по порядку из каждой комнаты. Как только посетителей начинают выселять, значение семафора изменяется и разблокируются потоки-посетители, на место выселенных посетителей помещаются случайные посетители, которые прямо перед этим не были в гостинице.