

АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ.

ДОМАШНЕЕ ЗАДАНИЕ № 5

ВАРИАНТ № 12

Корепанова Полина Денисовна

БПИ207

Вариант№: 12.

Текст задания:

Задача о гостинице-2 (умные клиенты). В гостинице 10 номеров с ценой 200 рублей, 10 номеров с ценой 400 рублей и 5 номеров с ценой 600 руб. Клиент, зашедший в гостиницу, обладает некоторой суммой и получает номер по своим финансовым возможностям, если тот свободен. Если среди доступных клиенту номеров нет свободных, клиент уходит искать ночлег в другое место. Создать многопоточное приложение, моделирующее работу гостиницы.

Цели и задачи:

Изучить работу с потоками. Научиться разбивать задачу на части, для последующего их выполнения различными потоками.

В соответствии выданным вариантом задания было разработано консольное приложение с использованием стандартной библиотеки языка программирования C++. Программа компилируется и выполняется в ОС Linux.

Принцип работы программы:

Существует класс клиента отеля (Client), содержащий поле budget. Бюджет клиента является случайным числом и генерируется при появлении инстанса класса. Границы генерации бюджета – от 150 до 700. Если клиент не может позволить себе ни один из трёх типов номеров – выводится соответствующее сообщение. Клиенты входят в отель по одному и проходят через процесс регистрации, где определяется, какой номер им положен. После этого, при удачной регистрации клиента, он “отправляется в свой номер” (создаётся поток).

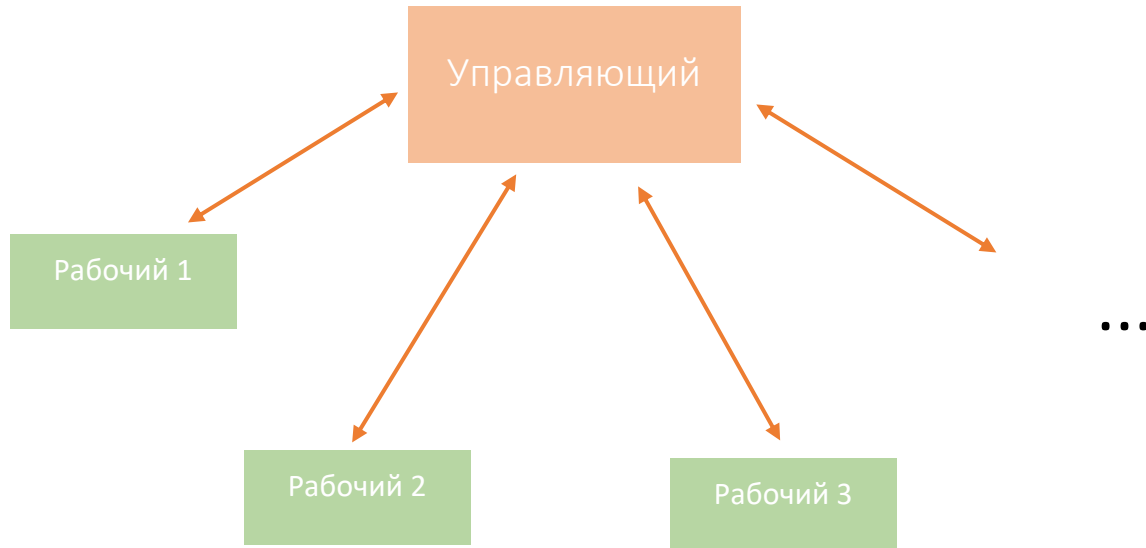
Для более наглядного представления в консоли, клиенты заходят в гостиницу с интервалом в одну секунду.

Так как данный тип задачи не даёт конкретных условий прекращения работы симуляции, примерное время работы программы можно изменить, записав в константу simulationTime соответствующее желаемому количеству минут. Значение по умолчанию – 1 минута.

Время пребывания клиентов в отеле (сек.) так же устанавливается в соответствующую константу stayingTime при необходимости. По умолчанию она содержит значение 20 – оптимальное для демонстрации.

Парадигма:

Процесс выполнения задачи был ориентирован на парадигму “Управляющий и рабочие”, где рабочие – потоки пребывания гостя в отеле.



Потоки, после выполнения выданной им задачи (в данном решении - сон), делают последнее изменение в общей памяти и заканчивают работу.

Ресурсы:

- [“Парадигмы параллельного программирования”](#)
- Cameron Hughes, Tracey Hughes, “Thread Models” – [Parallel and Distributed Programming Using C++] 20.02.2004 – (печатное издание)
- Sayan Mahapatra “Multithreading in C++” 12.10.21 - [GeeksForGeeks](#)