

Multithreading

Профессия Java-разработчик на Hexlet

Преподаватель: Яковлев Егор

Вопросы к лекции

- Что такое процесс?
- Что такое поток?
- Какие задачи решает многозадачность?

План

1. Многозадачность
2. Многопоточность
3. Возможности многопоточности
4. Поток и процесс
5. Проблемы многопоточности

Многозадачность

Многозадачность (multitasking) – свойство операционной системы или среды выполнения обеспечивать возможность параллельный (или псевдопараллельной) обработки нескольких задач

МНОГОПОТОЧНОСТЬ И МНОГОЗАДАЧНОСТЬ

По-настоящему параллельное выполнение задач возможно только в многопроцессорной системе, поскольку только в них присутствует несколько системных конвейеров для исполнения команд. В однопроцессорной многозадачной системе поддерживается так называемое псевдопараллельное исполнение, при котором создаётся видимость параллельной работы нескольких процессов. В таких системах, однако, процессы выполняются последовательно, занимая малые кванты процессорного времени

Какие задачи решает МНОГОПОТОЧНОСТЬ?

Многопотчность широко используется в приложениях с пользовательским интерфейсом. В этом случае за работу интерфейса отвечает один поток, а какие-либо вычисления выполняются в других потоках. Это позволяет пользовательскому интерфейсу не подвисать, когда приложение занято другими вычислениями

Какие задачи решает многопоточность?

Многие алгоритмы легко разбиваются на независимые подзадачи, которые можно выполнять в разных потоках для повышения производительности. Например, при фильтрации изображения разные потоки могут заниматься фильтрацией разных частей изображения

Какие задачи решает МНОГОПОТОЧНОСТЬ?

Если некоторые части приложения вынуждены ждать ответа от сервера/пользователя/устройства, то эти операции можно выделить в отдельный поток, чтобы в основном потоке можно было продолжать работу, пока другой поток ждёт ответа

Поток и процесс

Процесс – это абстракция, реализованная на уровне операционной системы.

Поток – это контейнер, в котором находятся ресурсы программы:

- адресное пространство
- потоки
- открытые файлы
- дочерние процессы

Процессы VS потоки

Процесс рассматривается ОС, как заявка на все виды ресурсов, кроме одного – процессорного времени.

Поток – это заявка на процессорное время. Процесс – это всего лишь способ сгруппировать взаимосвязанные данные и ресурсы, а потоки – это единицы выполнения, которые выполняются на процессоре

Процессы VS потоки

Поток легче, чем процесс, и создание потока стоит дешевле. Потоки используют адресное пространство процесса, которому они принадлежат, поэтому потоки внутри одного процесса могут обмениваться данными и взаимодействовать с другими потоками

Состояния потоков

- Выполняемый (Executing) – поток, который выполняется в текущий момент на процессоре
- Готовый (Runnable) – поток ждёт получения кванта времени и готов выполнять назначенные ему инструкции. Планировщик выбирает следующий поток для выполнения только из готовых потоков
- Ожидающий (Waiting) – работа потока заблокирована в ожидании блокирующей операции

Проблемы многопоточности: состояние гонки

Состояние гонки – ошибка проектирования многопоточной системы или приложения, при котором работа системы или приложения зависит от того, в каком порядке выполняются части кода

Проблемы многопоточности: **deadlock**

Deadlock – ситуация, при которой несколько потоков находятся в состоянии ожидания ресурсов, занятых друг другом, и ни один из них не может продолжить выполнение

Голодание потоков

Голодание потоков – это ситуация, в которой поток не может получить доступ к общим ресурсам, потому что на эти ресурсы всегда претендуют какие-то другие потоки, которым отдаётся предпочтение