

Учебное практическое задание 3

Тема Java Threads. Синхронизация потоков

Цель работы: Сформировать навыки работы с технологиями Java Threads.

Задание. Разработать многопоточное приложение.

Рекомендации по выполнению задания

1. Объявите и инициализируйте глобальные данные / переменные, которые требуют синхронизации (например, «count»). Объявите и инициализируйте связанный мьютекс.

Создайте потоки А и В для работы.

Поток А

- Работает до момента, когда должно произойти определенное условие (например, "счетчик" должен достичь указанного значения)
- Блокирует связанный мьютекс и проверяет значение глобальной переменной
- Вызывает `pthread_cond_wait()`, чтобы выполнить блокирующее ожидание сигнала от Thread-B. Обратите внимание, что вызов `pthread_cond_wait()` автоматически и атомарно разблокирует связанную переменную мьютекса, чтобы она могла использоваться Thread-B.
- При получении сигнала поток А просыпается, Мьютекс автоматически блокируется атомарно.

2. Поток В

- Меняет значение глобальной переменной, которую ожидает Thread-A.
- Проверяет значение глобальной переменной ожидания Thread-A. Если оно удовлетворяет желаемому условию, сигнализирует Thread-A.
- Разблокирует мьютекс.

3. Разработайте многопоточное приложение (использующее общую переменную):

- в потоке объявлена приватная переменная, например $i=0$. При выполнении потока i инкриминируется, к примеру, 100 раз. При каждом инкременте на консоль выводится значение i ;
- основная программа создает два потока. Проанализировать изменение i .

Как изменится поведение программы, если i объявить как локальную переменную (переменная метода `run()`)?

4. Разработайте многопоточное приложение:

- один поток считывает текст собственной программы;
- второй поток получает считанную строку, анализирует на вхождение образца и по условию вывод строку на экран.

5. Разработайте многопоточное приложение, реализующее кольцевой буфер в задаче «производитель-потребитель». Программа должна порождать процессы производители количеством $K>1$ и процессы потребители ($M>1$). Размер буфера N элементов. Частота работы производителя меньше в среднем в 10 раз частоты работы потребителя. Частоты задаются случайно. Производитель пишет в буфер в среднем 10 значений.

6. Предоставьте отчет, который содержит: текст программного кода выполненных заданий и скрин результатов работы. Приложите к отчету скомпилированные программы по выполнению указанных заданий.

При оформлении кода использование комментариев обязательно.