

Домашняя работа от 14.02

Взаимодействие потоков. Producer - Consumer

- Producer/Consumer. Реализуйте классы Store, Producer, Consumer с помощью `java.util.concurrent.Semaphore`

Семафор в данном примере используется для контроля доступа к общему ресурсу - количеству продуктов в магазине.

1. семафор для производителя. Производитель должен захватить этот семафор, чтобы увеличить количество продуктов в магазине. После добавления продукта производитель освобождает семафор потребителя, чтобы дать возможность потребителю забрать продукт.

2. семафор для потребителя. Потребитель должен захватить этот семафор, чтобы уменьшить количество продуктов в магазине. После забора продукта потребитель освобождает семафор продюсера, чтобы дать возможность производителю добавить новый продукт.

Таким образом, семафоры позволяют синхронизировать работу производителя и потребителя, обеспечивая правильное выполнение операций добавления и удаления продуктов из магазина.

- Задача о читателях и писателях. К базе данных имеют доступ два типа процессов - читатели и писатели. Первые могут выполнять транзакции, которые могут только читать данные, для вторых добавляется возможность их читать. Необходимо обеспечить такой доступ к базе данных, что в любой момент времени в ней находится либо один писатель и ни одного читателя, либо сколь угодно читателей, но ни одного писателя.

База данных в этом примере условная - ей может выступать какой-то ресурс (массив)

Date Time API

- Реализовать метод `isDateOdd(String date)` так, чтобы он возвращал `true`, если количество дней с начала года - нечетное число, иначе `false`

Пример: "JANUARY 1 2000" => true "JANUARY 2 2020" => false

- Распечатать график праздников на текущий год используя форматирование.

- Интернет-магазин

- Создать класс Товар, имеющий переменные имя, цена, рейтинг.

- Создать класс Категория, имеющий переменные имя и массив товаров. Создать несколько объектов класса Категория.

- Создать класс Basket, содержащий массив купленных товаров.