

Blatt 8

Betriebssysteme

Luis Staudt

Aufgabe 1: Körner Zähler (KoernerZaehler)

Frage 1.1: Wie erhält man die Körnerzahl der aktuellen Kachel?

Antwort: `gibTerritorium().gibKoerner(gibPosition())`

Frage 1.2: Wie wird das Ergebnis aus einem Future-Objekt geholt?

Antwort: `future.get()`

Aufgabe 2: Größte Körnerzahl (KoernerMaximal)

Frage 2.1 Hier fehlt ein Attribut zur Synchronisation der Sucher

Antwort: `private CountDownLatch latch;`

Frage 2.2: Hier fehlt ein Parameter zur Synchronisation der Sucher

Antwort: `CountDownLatch latch`

Frage 2.3: Warum ist die aktive Warteschleife schlecht?

Antwort: Verschwendet CPU-Zyklen unnötig - `CountDownLatch.await()`

ist effizienter und blockiert den Thread passiv.

Aufgabe 3: Körner Tauscher (KoernerTauscher)

Frage 3.1: Mit welcher Methode wird die Körnerzahl ausgetauscht?

Antwort: `austauscher.exchange(meineKoernerZahl)`

Frage 3.2: Wie wird die neue Körnerzahl gesetzt?

Antwort: `gibTerritorium().setzeKoernerZahl(gibPosition(), andereKoernerZahl)`

Aufgabe 4: Körner Esser (KoernerEsser)

Frage 4.1: Hier fehlt ein Parameter im Konstruktor - welcher?

Antwort: `CyclicBarrier barriere`

Frage 4.2: Wieviele Threads müssen es sein? Warum dürfen es nicht weniger sein als die Zahl der Esser?

Antwort: Mindestens so viele wie Esser. Grund: Alle Esser müssen gleichzeitig an der CyclicBarrier warten können, sonst Deadlock.

Frage 4.3: Bekommt jeder KoernerEsser seine eigene CyclicBarrier oder dieselbe?

Antwort: Dieselbe - nur so können sie sich synchronisieren.

Frage 4.4: Mit welchem Wert muss die CyclicBarrier erzeugt werden?

Antwort: Mit der Anzahl der Esser (`zahlEsser`)

Frage 4.5: Hier fehlt ein Übergabeparameter - welcher?

Antwort: `CyclicBarrier barriere`

Frage 4.6: Mit welcher Methode von Territorium kann man die Zahl der Körner um eins verringern?

Antwort: `dekrementiereKoerner(int[] position)`

Frage 4.7: Wie kann man die CyclicBarrier verwenden, damit jeder Esser die ursprüngliche Zahl der Körner sieht?

Antwort: `barriere.await()` vor dem Lesen der Körnerzahl aufrufen