

Kodieren Sie die Klasse *Schlucht* die folgendes Problem, ohne Deadlock und unbestimmtes Aufschieben löst:

Ueber eine Schlucht, die in Richtung Nord-Süd verläuft ist ein Seil gespannt. Man kann diese Schlucht nur überqueren, falls man sich unter Verwendung dieses Seils (z.B. hinüberhangeln) von der einen auf die andere Seite begibt. Es ist denkbar, dass gleichzeitig Personen von West nach Ost und von Ost nach West die Schlucht überqueren wollen. Aus Sicherheitsgründen darf man nicht kreuzen auf dem Seil. Falls in einer Richtung Personen unterwegs sind und auf der Gegenseite eine oder mehrere Personen warten, so dürfen aus dieser Richtung keine neuen Personen aufs Seil bis mindestens eine Person in der Gegenrichtung die Schlucht überquert hat. Kodieren Sie die Methoden `EastWest` und `WestEast` unter Verwendung von Condition Variables und Locks aus `java.util.concurrent`.

Was müsste vorgekehrt werden für Deadlock Verhinderung?