

Παράλληλος και Κατανεμημένος Υπολογισμός

Εργαστήριο 4

Αμοιβαίος αποκλεισμός

Πρώτα μελετήστε τις σημειώσεις. Στη συνέχεια ξεκινήστε από τους κώδικες Code/SharedCounter, SharedCounterMutex και SharedCounterSync.

Βεβαιωθείτε ότι καταλαβαίνετε τους μηχανισμούς εκτέλεσης στους κώδικες SharedCounter και τους λόγους που το αποτέλεσμα δεν είναι το αναμενόμενο.

Βεβαιωθείτε ότι καταλαβαίνετε τους μηχανισμούς εκτέλεσης στους κώδικες αμοιβαίου αποκλεισμού στα SharedCounterMutex και SharedCounterSync, και τους λόγους που το αποτέλεσμα είναι το αναμενόμενο.

Εφαρμόστε τις τεχνικές των κωδίκων Code/SharedCounterMutex και SharedCounterSync στους κώδικες Code/SharedCounterExercise.

1. Οι κώδικες Code/SharedCounterExercise χρησιμοποιούν καθολικές μεταβλητές. Γράψτε τους ίδιους κώδικες αλλά με χρήση ορισμάτων του main(). Η χρήση αντικειμένων προαιρετική.

Εκτελέστε τους κώδικες και βεβαιωθείτε ότι καταλαβαίνετε τα αποτελέσματα.

2. Εφαρμόστε τις τεχνικές των κωδίκων Code/SharedCounterMutex στους κώδικες Code/SharedCounterExercise.

Εκτελέστε τους κώδικες και βεβαιωθείτε ότι καταλαβαίνετε τα αποτελέσματα.

3. Εφαρμόστε τις τεχνικές των κωδίκων Code/SharedCounterSync στους κώδικες Code/SharedCounterExercise.

Εκτελέστε τους κώδικες και βεβαιωθείτε ότι καταλαβαίνετε τα αποτελέσματα.

4. Στη σελ 54 στο Notes_04.pdf προτείνεται μια απλή μέθοδος για την αύξηση της διακριτότητας του **synchronized**. Γράψτε ένα πρόγραμμα **DoubleCounterSync.java** όπου να υπάρχει μια κλάση **DoubleCounter** που να περιέχει δύο ακεραίους μετρητές **n1** και **n2**, οι οποίοι να μπορούν να αυξάνονται ταυτόχρονα από πολλά νήματα, ανεξάρτητα ο ένας από τον άλλο.