

## Παράλληλος και Κατανεμημένος Υπολογισμός

### Εργαστήριο 5

#### 1. Μελετήσετε τον 91\_Code/ConcurrentCollections.

Εκτελέστε τους κώδικες με διαφορετικές καθυστερήσεις ή άλλες δικές σας τροποποιήσεις για να δείτε τη λειτουργία τους.

Οι κώδικες στο folder ConcurrentCollections είναι γραμμένοι `πρόχειρα`. Οργανώστε καλύτερα τον διαμοιρασμό και τον αμοιβαίο αποκλεισμό. Δοκιμάστε λύσεις

- (α) με μοιραζόμενες δομές και αντικείμενα και
- (β) με και Lock/Semaphores και synchronized

#### 2. Μελετήστε και τους κώδικες 91\_Code/Atomic, BankAccount (και 05\_Code/BankAccount)

Ο κώδικας BankAccount αναφέρεται σε ένα αντικείμενο. Δημιουργείστε μια δομή από αντικείμενα BankAccount. Θεωρείστε ότι έχετε δύο ειδών νήματα.

(α) Νήματα ανάγνωσης, που διαβάζουν ένα τυχαίο λογαριασμό, διαφορετικό κάθε φορά. Τα νήματα αυτά είναι πολλά, εκτελούν συχνές και πολλές επαναλήψεις και η εκτέλεσή τους πολύ σύντομη.

(β) Νήματα εγγραφής, που τροποποιούν ένα τυχαίο λογαριασμό, διαφορετικό κάθε φορά. Τα νήματα αυτά είναι λίγα, εκτελούν αραιές και λίγες επαναλήψεις και η εκτέλεσή τους πολύ αργή (βάλτε σχετικά μεγάλη καθυστέρηση).

Δοκιμάστε διάφορους τύπους αμοιβαίου αποκλεισμού και μετρήστε τις επιδόσεις τους.

#### 3. Γευματίζοντες Φιλόσοφοι

Μελετήστε τον κώδικα 05\_Code/DiningLock

Δοκιμάστε λύσεις (κυρίως αποτροπής του κύκλου αναμονής). Αν θέλετε δοκιμάστε και διαφορετικούς τύπους αμοιβαίου αποκλεισμού.