

## Παράλληλος και Κατανεμημένος Υπολογισμός

### Εργαστήριο 4

1. Μελετήστε τον κώδικα Code/ProdCons. Εκτελέστε τον με διάφορα σενάρια (αριθμό παραγωγών, καταναλωτών, μέγεθος buffer, καθυστερήσεις).

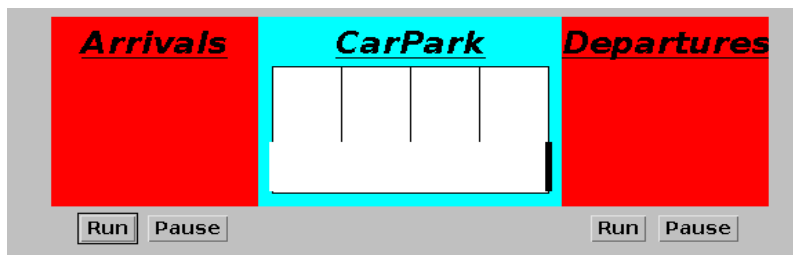
Με βάση τους κώδικες ProdConSemMux, ProdConLockCond και ProdConSyncCond παράγετε λύσεις με χρήση Σηματοφόρων, Κλειδωμάτων και Μεταβλητών Συνθήκης, και Εποπτών για μέγεθος buffer, δηλαδή size = 1 και για έναν παραγωγό και ένα καταναλωτή (SPSC).

Οι γενικές λύσεις δουλεύουν αλλά **ζητούνται ειδικές λύσεις** που αφαιρούν όλες τις περιττές γραμμές κώδικα (δηλαδή τους περιττούς μηχανισμούς ταυτοχρονισμού) για την κάθε περίπτωση. Στόχος της άσκησης είναι να ελέγξετε αν έχετε κατανοήσει τον ρόλο του κάθε μηχανισμού ταυτοχρονισμού.

2. Με βάση τον κώδικα BlockingQueue που επισυνάπτεται δημιουργήστε ένα αντίγραφο του κώδικα Code/ProdCons όπου θα αντικαταστήσετε το Buffer με ένα ArrayBlockingQueue.

*Σημείωση: Όσοι/ες χρησιμοποιούν άλλη γλώσσα ας ψάξουν να βρούν αντίστοιχη δομή. Πχ στη Python υπάρχει η Queue.*

3. Ολοκληρώστε τον σκελετό κώδικα που CarPark που επισυνάπτεται. Το σενάριο παρακάτω:



Έχουμε ένα parking που διαθέτει capacity θέσεις (στην εικόνα capacity = 4). Επίσης έχουμε έναν αριθμό αυτοκινήτων που φθάνουν να παρκάρουν σε τυχαία χρονικά διαστήματα (arrivals). Όταν βρούν θέση παρκάρουν (parking) και μετά από τυχαίο χρόνο αναχωρούν (departures).

Στον σκελετό του κώδικα που σας δίνεται λείπουν οι μηχανισμοί ταυτοχρονισμού. Έτσι οι εκτυπώσεις δεν εμφανίζονται με τη σειρά, η μεταβλητή spaces που μετράει τις άδειες θέσεις δεν προστατεύεται, η είσοδος και η έξοδος δεν συγχρονίζονται.

Γράψτε τρεις κώδικες ParkSem.java, ParkLockCond, ParkMon.java με χρήση σηματοφόρων, κλειδωμάτων και μεταβλητών συνθηκών, επόπτη Java, που να λύνουν τα παραπάνω προβλήματα.

*Σημείωση 1. Όσοι/ες χρησιμοποιούν άλλη γλώσσα πιθανότατα δεν θα μπορέσουν να υλοποιήσουν την λύση επόπτη.*

*Σημείωση 2. Όσοι/ες θέλουν μπορούν να επεκτείνουν τον κώδικα ώστε να εμφανίζει όχι μόνο το πλήθος των άδειων θέσεων αλλά τις ακριβείς θέσεις, που παρκάρει κάθε αυτοκίνητο και ποιες θέσεις είναι κενές.*