

Σειρά Εργασιών 2

2.1 Δυαδικοί σηματοφόροι

Υλοποιήστε, ως ξεχωριστό τμήμα λογισμικού στο πνεύμα μιας βιβλιοθήκης, τους δικούς σας δυαδικούς σηματοφόρους. Η υλοποίηση σας πρέπει να παρέχει τις λειτουργίες `mysem_create()`, `mysem_down()`, `mysem_up()`, `mysem_destroy()`, και να υποστηρίζει την δημιουργία/χρήση ξεχωριστών, ανεξάρτητων σηματοφόρων (σκεφτείτε πως πρέπει να οριστούν οι παράμετροι και τιμές επιστροφής κάθε λειτουργίας). Η υλοποίηση σας πρέπει να χρησιμοποιεί τους γενικούς σηματοφόρους του `system V`, που μπορείτε να υποθέσετε ότι είναι δίκαιοι. Επίσης, η υλοποίηση πρέπει να εντοπίζει χαμένες κλήσεις της `mysem_up()`, στην οποία περίπτωση να επιστρέφει κωδικό λάθους στην εφαρμογή.

Στις υπόλοιπες εργασίες, ο συγχρονισμός θα υλοποιείται με βάση τους δικούς σας σηματοφόρους.

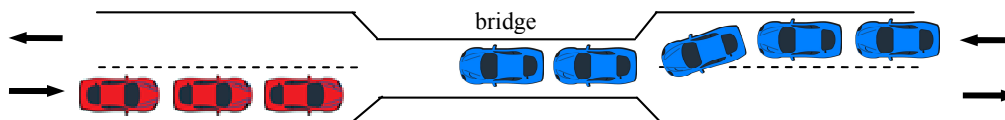
2.2 Αναγνώριση πρώτων αριθμών

Αλλάξτε το πρόγραμμα που αναπτύξατε στην εργασία 1.2 έτσι ώστε ο επιθυμητός συγχρονισμός ανάμεσα στα νήματα να γίνεται αποκλειστικά μέσω δυαδικών σηματοφόρων.

2.3 Στενή γέφυρα

Αναπτύξτε κώδικα ελέγχου για την ρύθμιση της κυκλοφορίας οχημάτων πάνω από μια στενή γέφυρα, έτσι ώστε να διασφαλίζονται οι εξής ιδιότητες:

- Δεν επιτρέπεται να υπάρχουν πάνω στη γέφυρα οχήματα που κινούνται προς αντίθετες κατευθύνσεις.
- Δεν επιτρέπεται να υπάρχουν πάνω στη γέφυρα περισσότερα από N οχήματα.
- Δεν επιτρέπεται ένα όχημα να περιμένει για πάντα να περάσει την γέφυρα, ακόμα και αν καταφθάνουν συνεχώς οχήματα στην άλλη πλευρά της γέφυρας.



Υλοποιήστε τον επιθυμητό συγχρονισμό ανάμεσα στα οχήματα, μέσω κατάλληλου κώδικα «εισόδου» που εκτελεί κάθε όχημα όταν φτάνει στην γέφυρα, και κώδικα «εξόδου» που εκτελεί κάθε όχημα αφού περάσει τη γέφυρα. Ο συγχρονισμός πρέπει να γίνεται απ' ευθείας ανάμεσα στα οχήματα, χωρίς να υπάρχει κάποιο επιπλέον νήμα που παίζει τον ρόλο του μεσάζοντα/τροχονόμου. Η υλοποίηση σας πρέπει να βασίζεται αποκλειστικά σε δυαδικούς σηματοφόρους. Για να δοκιμάσετε/επιδείξετε την λύση σας, αναπτύξτε ένα απλό πρόγραμμα προσομοίωσης που δημιουργεί νήματα/οχήματα σε κάθε πλευρά της γέφυρας με βάση πληροφορία που διαβάζει από την είσοδο του.

2.4 Τρενάκι

Το τρενάκι ενός λούνα-παρκ χωράει N επιβάτες. Το τρενάκι αρχίζει την επόμενη διαδρομή μόνο όταν γεμίσει. Οι επιβάτες αποβιβάζονται από το τρενάκι όταν αυτό ολοκληρώσει την τρέχουσα διαδρομή, προτού αρχίσει η επιβίβαση των επόμενων επιβατών.



Υλοποιήστε τον επιθυμητό συγχρονισμό ανάμεσα στους επιβάτες και το τρενάκι, χωρίς κάποιο άλλο βοηθητικό νήμα. Η υλοποίηση σας πρέπει να βασίζεται αποκλειστικά σε δυαδικούς σηματοφόρους. Για να δοκιμάσετε/επιδείξετε την λύση σας, αναπτύξτε ένα απλό πρόγραμμα προσομοίωσης που δημιουργεί ένα νήμα για το τρενάκι και επιπλέον νήματα-επιβάτες με βάση πληροφορία που διαβάζει από την είσοδο του.