

## Σειρά Εργασιών 2

### 2.1 Αγωγοί σε επίπεδο χρήστη/συστήματος (δεν παραδίδεται/εξετάζεται)

Τροποποιήστε την υλοποίηση του αγωγού που αναπτύξατε στην εργασία 1.1 έτσι ώστε να γίνεται εθελοντική παραχώρηση του επεξεργαστή (μέσω `pthread_yield`) όσο ισχύει η συνθήκη αναμονής. Συγκρίνετε την απόδοση της υλοποίησης σας με αυτήν των αγωγών του συστήματος, π.χ., για την μετάδοση 1Mbytes ανάμεσα σε δύο νήματα, σε τμήματα 1 byte. Τι παρατηρείτε και γιατί;

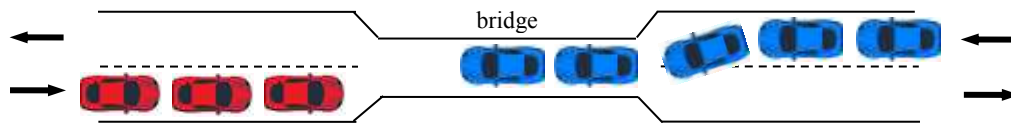
### 2.2 Υπολογισμός fractals χωρίς ενεργή αναμονή

Αλλάξτε το πρόγραμμα που αναπτύξατε στην εργασία 1.2 έτσι ώστε ο συγχρονισμός μεταξύ του κυρίως νήματος και των νημάτων εργάτη να επιτυγχάνεται χωρίς ενεργή αναμονή.

### 2.3 Στενή γέφυρα

Αναπτύξτε κώδικα ελέγχου για την ρύθμιση της κυκλοφορίας 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 παραπάνω από N οχήματα.
- Από τη στιγμή που ένα όχημα περιμένει να διασχίσει τη γέφυρα, κάποια στιγμή θα το καταφέρει, ακόμα και αν η κίνηση από το αντίθετο ρεύμα είναι συνεχής.



Υλοποιήστε τον επιθυμητό συγχρονισμό ανάμεσα στα οχήματα, με την μορφή κώδικα «εισόδου» που εκτελεί κάθε όχημα όταν φτάνει στην γέφυρα, και κώδικα «εξόδου» που εκτελεί κάθε όχημα όταν περάσει τη γέφυρα. Δεν πρέπει να υπάρχει περίπτωση ενεργής αναμονής. Για να δοκιμάσετε/επιδείξετε την λύση σας, αναπτύξτε ένα απλό πρόγραμμα προσομοίωσης που δημιουργεί νήματα/οχήματα με βάση πληροφορία που διαβάζει από την είσοδο του.

### 2.4 Τρενάκι

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



Υλοποιήστε τον επιθυμητό συγχρονισμό ανάμεσα στους επιβάτες και το τρενάκι, χωρίς ενεργή αναμονή. 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.

**Σημείωση για όλες τις παραπάνω εργασίες:** Η υλοποίηση πρέπει να γίνει σε C με χρήση της βιβλιοθήκης `pthread`. Ο συγχρονισμός μεταξύ των νημάτων πρέπει να υλοποιηθεί χρησιμοποιώντας `mutexes` και τις λειτουργίες `mutex_init`, `mutex_lock` και `mutex_unlock`, στο πνεύμα δυαδικών σηματοφόρων.