

R E A D M E  
A L G O R I T M I   P A R A L E L I   S I   D I S T R I B U I T I  
T E M A   2

Calmis Alina 335CB

`simple_semaphore:`

Pentru implementarea primului exercitiu am folosit un simplu semafor, fara a crea vreo clasa.

Pentru exercitiile urmatoare exista cate o clasa aferenta fiecarui tip de intersectie.

`simple_n_roundabout:`

Clasa SimpleNRoundAbout contine datele aferente din fisierul de input , cat si un semafor cu numarul de permits egal cu ultimul numar din fisier. Implementarea este asemanatoare exercitiului precedent.

`simple_strict_1_car_roundabout:`

Clasa SimpleStrictOneCar contine, la fel, pe langa datele din fisier, un vector de semafoare, care permite trecerea unei singure masini intr-o directie. Implementarea in IntersectionHandlerFactory , se diferentiaza de celelalte , prin faptul ca se alege intial semaforul din directia de start a masinii.

`simple_max_x_car_roundabout:`

Clasa SimpleMaxXCar contine , pe langa datele de input, un vector de semafoare, de data aceasta initializat cu numarul de permits-uri egal cu X din fisier.

`Crosswalk:`

Clasa Crosswalk continue un sinur ArrayList care retine ce culoare a fost ultima atribuita fiecarei masini. In intersectionHandlerFactory, atat timp cat nu s-a finisat, verifica daca pietonii trec, aunci seteaza rosu, cand pietonii stau seteaza verde. Daca toti pietonii au trecut, verifica sa nu mai ramana la sfarsit masini setate pe rosu.