Final exam of algorithms and data structures 2022-2023

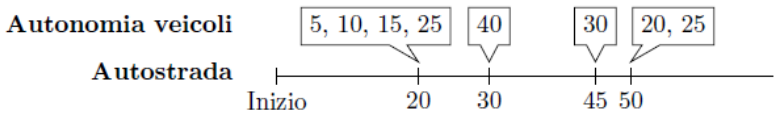
Consider a motorway described as a sequence of service stations. Each service station is located at a distance from the start of the motorway expressed in kilometres by a positive or null integer. There are no two petrol stations with the same distance: each petrol station is therefore uniquely identified by its distance from the start of the motorway.

Each service station is equipped with a fleet of electric vehicles for hire. Each vehicle is distinguished by the autonomy given by a battery charge, expressed in kilometres, by a positive integer number. The fleet vehicles of a single station contain a maximum of 512 vehicles. Having rented a car from a station s, it is possible to reach all stations whose distance from s is less than or equal to the car's autonomy.

A route is identified by a sequence of stations at which the driver stops. It therefore begins at one service station and ends at another, passing through zero or more intermediate stations. Assume that the driver hires a new car each time he stops at a service station s: the driver is then able to reach any service station at a distance from s less than or equal to than the car with the maximum available autonomy in s.

The objective of the project is as follows: given a pair of stations, plan the route with the least number of stops between them. If there are several routes with the same minimum number of stages the route with the shortest distance to the start of the motorway should be chosen.

Example:



In this example, the correct path between the station at distance 20 and the station at distance 50 is 20 → 30 → 50 (and not 20 → 45 → 50). Note that, on the other hand, 50 → 30 → 20 is the correct path between the station at distance 50 and the station at distance 20 (thus in the direction from right to left).

Input file format and expected printouts

The input text file contains a sequence of commands, one per line, with the following format. All values positive or null integers are encodable in 32 bits.

- add-station distance number-cars autonomy-car-1 ... autonomy-car-n
  - Adds a station, located at the indicated distance, with the indicated range number-auto.
  - For example: add-station 10 3 100 200 300, adds a station at distance 10 from the start of the motorway, with a fleet of three vehicles with an autonomy of 100, 200 and 300 km respectively. If a station already exists at the specified distance, the command does nothing.
  - Print expected as answer: added / not added.

- demolish-station distance
  - Removes the station at the indicated distance, if it exists.
  - Print expected as answer: demolished / not demolished.

- add-car distance-station autonomy-car-to-add
  - If the station exists, adds a car to it. It is possible to have several cars with the same autonomy.
  - Print expected as answer: added / not added.

- scrap-car distance-station autonomy-car-to-scrap
    - Removes a car from the indicated station if the station exists and has at least one car with the indicated autonomy.
    - Print expected as answer: scrapped / not scrapped.

- plan-route distance-station-departure distance-station-arrival
    - Requests to plan the route with the constraints indicated above.
    - Print expected response:
        - the stages in order of travel, represented with the distance of the stations from the start of the motorway, separated by spaces and at the end followed by an a-heading. Departure and arrival; if they coincide, the station is printed only once.
        - If the route does not exist, print: no route.
        - The planning action does not alter the stations or their fleet. The given stations are definitely present.