Heist to the museum*

A group of *M thieves* plans to steal the paintings in exhibition at a museum in Aveiro. The paintings are in display in *N* different rooms, each having Q_i paintings hanging on the walls, with $i = 0, 1, \dots, N\text{-}1$.

The operation is directed by the *master thief* who organizes her companions, standing in queue while waiting to be summoned, in assault parties of *K* elements and assigns to each party a target room in the museum. Different parties may proceed to different rooms at the same time. In the end, the master thief calls a meeting and informs the rest of the thieves about the sack earnings.

To prevent detection by the museum guards, the thieves forming a party crawl in line, as fast and as silently as they possibly can, along a previously established path between the outside gathering site and the target room. They must ensure that contiguous elements in the crawling line are not separated by a distance larger than a previously fixed value. Upon arrival at a museum room, each party member looks for a painting still hanging on the wall and, if there is one, he takes it down, detaches the canvas from the framing, rolls it over and inserts it in a cylinder container he carries on his back. He then prepares to leave the room. The way out is the same as the way in and the crawling procedure adopted before is adopted again. When a thief reaches the outside location where the master thief is hiding, he takes the canvas out of the cylinder and hands it to the master thief who stores it in the back of a van, or tells her he is coming empty-handed. Since the master thief does not know beforehand how many paintings are hanging in the walls of each room, she goes on promoting incursions to the same room until she is sure the room is empty.

The crawling movement of party G_j, with $j = 0, 1, \dots, (M\text{-}1)/K\text{-}1$, requires successive increments of position that obey the following rules

- the ingoing movement (from the outside gathering site to the museum room) is performed by taking positive position increments and the outgoing movement (from the museum room to the outside gathering site) is performed by taking negative position increments
- the distance between the outside gathering site and the museum room *i* is D_i length units, with $i = 0, 1, \dots, N\text{-}1$
- the ingoing movement only starts when all group members have been selected and are ready to proceed, the outgoing movement only starts when all group members have taken a canvass or are empty-handed, because no more paintings are hanging in the room walls
- the thieves in a party crawl in line, can overtake one another, but can never stay side by side, nor be separated by a distance larger than *S* length units
- at each iteration step, the thief t_j, with $j = 1, \dots, M\text{-}1$, can change his position from 1 to MD_{t_j} length units, always moving as fast as he possibly can without violating the constraints imposed by the previous rule
- the maximum displacement, MD_{t_j}, is specific to each thief t_j, the thieves are not all equal, some are more agile and faster than others.

Assume there are 7 thieves in the whole, master included, the maximum displacement of the ordinary thieves is a random number between 2 and 6, the number of exhibition rooms having paintings in display is 5 with random distances to the outside concentration site between 15 and 30, the number of paintings hanging in each room is a random number between 8 and 16 and that the assault parties have 3 elements. Also assume that the maximum separation limit between thieves crawling in line is 3 length units.

Write a simulation of the life cycle of the thieves using the client-server model with server replication, where the thieves are the *clients* and the access to the information sharing regions are the services provided to them by the *servers*.

The operations that were previously assigned to activities carried out in the information sharing regions (for the already implemented concurrent version), must now be assigned to independent requests performed on the servers through message passing. In each case, a connection has to be established, a request has to be made, waiting for the reply will follow and the connection has to be closed.

One aims for a solution to be written in Java, to be run in Linux under TCP sockets, either in a concentrated manner (on a single platform), or in a distributed fashion (up to 7 different platforms), and to terminate (it must contemplate service shutdown). A *logging* file, that describes the evolution of the internal state of the problem in a clear and precise way, must be included.

* Concept by Pedro Mariano

Guidelines for solution implementation

1. Specify for each representative server of an information sharing region the structure of the messages to be exchanged.
2. Specify the general organization of the servers architecture.
3. Specify the general organization of the clients architecture.
4. Sketch the interaction diagram which describes in a compact, but precise, way the dynamics of your solution. Go back to steps 1, 2 and 3 until you are satisfied the description is correct.
5. Proceed to its coding in Java as specific reference data types.
6. Specify the mapping of the servers and the clients onto multiples nodes of the parallel machine and write the shell scripts which enable the deployment and the execution of the different modules the application is composed of.
7. Validate your solution by taking several runs and checking for each, through the detailed inspection of the logging file, that the output data is indeed correct.