Parallel Processing – task 1

15.10.2015

Design and implement the synchronization protocol (based on semaphores) for the following problem.

Problem

1. There are three mines: Lead Mine, Mercury Mine and Sulfur Mine. Each of them has three Representatives who sell their products in the local market.
2. Each of the Representatives comes in random intervals (from 200 to 500ms) and tries to show his goods in the marketplace.
3. There are just 2 stalls for each good in the marketplace, so the Representatives may have to wait until there is a place for them.
4. There are three guilds of Alchemists trying to obtain gold: $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$.
5. Alchemists $\mathcal{A}$ try to obtain gold from lead and mercury.
6. Alchemists $\mathcal{B}$ try to obtain gold from mercury and sulfur.
7. Alchemists $\mathcal{C}$ try to obtain gold from mercury, sulfur and lead.
8. Alchemists come to the market to buy the ingredients they need. They come in random order. There is some (random) interval between the appearances of two Alchemists.
9. Each Alchemist tries to collect all necessary ingredients. If they are not available, the Alchemist waits.

10. After collecting all the ingredients, each Alchemist goes to his laboratory (and therefore leaves the marketplace).

Technical aspects

1. The solution should offer the maximum parallelism. The following situation is erroneous:

   There is some mercury and sulfur available (but no lead). There are some Alchemists $\mathcal{B}$ waiting, but they are blocked by the Alchemist $\mathcal{C}$, who came to the marketplace earlier.

2. You should avoid the starvation of Alchemists $\mathcal{C}$.

3. The solution when the Alchemist takes some ingredients and returns them (if not everything he needs in available) is not the best possible (and therefore it is not worth maximum number of points).

4. Any kind of busy waiting (even a little one) is wrong.

5. The synchronization should use only simple semaphores and methods `acquire()` and `release()`.

Deadline for submitting your solution: 03.11.2015 (23:59)