Parallel computing — 19.1.2018
Exercise 2

Submit the solutions to Moodle by 19.1. 09:00 (1 hour before exercises).

Kuopio group may be held on Monday 22.1. at 10-12, we'll agree later.

6. Let us assume that N persons (e.g., 32) sit in a circle, each person having a display (or a paper or a blackboard) on which only (s)he can write and all others can see it simultaneously. Moreover, each person has a unique ID (1..N), and all persons know all IDs. This essentially formulates a CROW PRAM. The persons may have some local memory (in their minds), but the others can not see it. The displays can contain at most 10 integers at a time (at the beginning, the inputs take one position). The persons may not speak or otherwise communicate together, just use the displays.

   At the beginning, all persons have an integer (0-99) in their displays. Write an algorithm for counting the sum of all values. At the end of the algorithm the result must be on displays of all persons. Estimate also the execution time of your algorithm (in steps and seconds).

   Write the algorithm so that it is can be executed with any set of average humans (non-computing professional, but literate and calculation-able). Use statements like "add the number on display ID+1 to your own number and show it on your display".

   Write an algorithm that works in $o(N)$ addition steps (note: little-o, i.e., asymptotically less than N). Thus you may not use algorithm: "everyone reads all values, counts the sum, and displays the result". Remember also that N/c is not $o(N)$.

7. Let us modify the rules of the task 6 by limiting each display to contain only 1 integer at a time. The persons may not speak to each other, but they can raise and lower their left hands (and, obviously, see the hands of other people). You can assume the any person can quickly inspect whether all hands are down or not.

8. Again use the rules of the task 6. Now make an algorithm for voting. Each person votes either "yes", "no", or "neither". The task is to find are there more "yes" -votes or "no" -votes.

9. What are the time complexity, used number of processors, and work of the following parallel algorithm (as a function of N)? The computation result should be the same as if all loops were sequential. Which PRAM variant is required? How to modify the algorithm to be executed in an EREW PRAM.

```
for i := 1 to N pardo
    for j := 1 to N pardo
        A[i, j] := 1;
for i := 1 to N do
    for j := 1 to N pardo
        for k := 1 to N pardo
            A[i, j] := A[i, j] + k;
```

10. Let us assume that we are building a PRAM with 1000 processors at 5 GHz each. There is an interconnection network that connects all processors to all memory modules. Each processor reads or writes a 64-bit word every fifth clock cycle. We'll ignore latencies and collisions. How much bandwidth is needed for the memory modules (and at the interconnection network between the processors and memory modules)? How many memory chips are needed if each chip provides bandwidth of 1 GB/s (block access)?