06/01/2025

Distributed Algorithms - 83453 - Homework 3

due: 20/01/2024

1) Define the *k-set consensus problem* as follows.
   Each processor starts with an input $x_i \in \mathbb{Z}$, and outputs a value y_i and it holds that:

   a. Validity: $y_i \in \{x_1, \ldots, x_n\}$

   b. k-Agreement: $\left|\{y_i \mid 1 \le i \le n\}\right| \le k$,
   that is, the set of outputs has at most k different values

   Prove that the following algorithm is **f-resilient** to crash-failures.

Algorithm 18 k-set consensus algorithm in the presence of crash failures: code for processor p_i, $0 \le i \le n-1$.

Initially $V = \{x\}$

round r, $1 \le r \le \frac{f}{k} + 1$: // assume that k divides f
   send V to all processors
   receive S_j from p_j, $0 \le j \le n-1$, $j \ne i$
   $V := V \cup \bigcup_{j=0}^{n-1} S_j$
   if $r = f/k + 1$ then $y := \min(V)$ // decide

---

2) Assume a byzantine-consensus with $n=4$ parties and $f=1$ byzantines. Assume the inputs of the parties belong to the set {0,1,2,3}.

   a. Show that no consensus algorithm satisfies "non-faulty-node input Validity" (as defined in Slide 37 in Presentation 6)

   b. Now assume that the inputs and outputs must be form the set {0,1}. Prove that "non-faulty-node input Validity" (as defined in Slide 37 in Presentation 6) is equivalent to "all-same validity" (Slide 38, Presentation 6).

3) Assume a ring network with $n \gg 10$ nodes. We would like to "split" the ring into consecutive segments, where each segment contains at least 5 neighboring nodes, but no more than 10 nodes.

Specifically, nodes have no inputs (but have IDs), and each node needs to end up at one "segment" and know to which segment it belongs to.

a. Design a synchronous deterministic algorithm that solves the above task in $O(\log^* n)$ time. Explain why your algorithm works correctly and analyze its time complexity.

b. Prove that without IDs, this task cannot be solved by a synchronous deterministic algorithm.