ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

FS 2009

Prof. R. Wattenhofer
C. Lenzen, T. Locher

Exam
Principles of Distributed Computing

03.08.2009

Do not open or turn until told so by the supervisor!

Notes

There is a total of 120 points. The number of points is given before each individual question in parentheses. The total for each group of questions is indicated after the title.

Your answers may be in English or in German. Algorithms can be specified in high-level pseudocode or as a verbal description, unless otherwise mentioned. You do not need to give every last detail, but the main aspects need to be there. Big-O notation is acceptable when giving algorithmic complexities. However, try to provide exact bounds whenever possible.

Points

Please fill in your name and student ID before the exam starts.

Name	Legi-Nr.

Question Nr.	Achieved Points	Max Points
1		24
2		32
3		33
4		31
Total		120

1 Distributed Computation Models (24 Points)

a) (3) When is it reasonable to assume that local computations take zero time?

b) (3) Give one example each for distributed systems whose communication can be modeled by means of message passing, shared memory, or all-to-all communication. Add short explanations for your choices.

c) (2) Do you consider a running time in the order of the graph diameter fast or slow in the message passing model? Explain your reasoning!

d) (2) In the message passing model, when does it *not* matter whether communication is synchronous or asynchronous?

e) (3) In the shared memory model, why does it affect the (potential) fault tolerance of the system whether communication is synchronous or asynchronous?

f) (2) In the all-to-all communication model, what prohibits the system from solving virtually everything in a single round?

g) (4) Explain the meaning of the notions of *online* and *offline* problems. Why might it be useful to analyze the offline complexity of a problem encountered in an online setting?

h) (2) What does "termination" mean for distributed algorithms?

i) (3) What is the relation between termination and self-stabilization?

2 Graph Problems (32 points)

Give distributed, synchronous, deterministic algorithms solving the following problems. You may employ any algorithm from the lecture fulfilling these criteria. Furthermore, the nodes have unique identifiers $1, \ldots, n$, n is known, and all graphs are connected.[1] No proofs are required in this exercise. (Note that you can still earn points if your solution is slightly worse than required!)

a) (3) A *dominating set* is a subset of the nodes such that any node is either in the set or has a neighbor in the set. On a ring, find in the least possible number of rounds a dominating set which is at most 3 times larger than the minimum size!

b) (5) Compute the diameter of an arbitrary graph in D time!

c) (5) Compute a 2-approximation on the diameter D of an arbitrary graph in $O(D)$ rounds, where messages contain at most $O(\log n)$ bits!

d) (3) Given a *bipartite*, i.e., 2-colorable graph, obtain a 2-coloring in at most D time using empty[2] messages!
Hint: Exploit that leader election is trivial due to the identifiers!

e) (8) Decide whether a graph is a tree in $O(n)$ rounds using at most $O(n)$ empty messages.

f) (8) The *girth* g of a graph is the size of a smallest cycle. Assume that g and all node degrees upper bounded by a constant. Provide an algorithm using messages of size $O(\log n)$ that terminates after at most $g + D$ rounds. Note that in the end all nodes must know g!

[1] Note, however, that the diameter D is unknown!
[2] That means a node either sends a message, or it does not, but does not contain any further information.

4 Minimum Dominating Sets (31 Points)

A *dominating set* is a subset of the nodes such that each node is either in the set or has a neighbor in the set. A *minimum dominating set* is a dominating set containing the least possible number of nodes. In the following, we denote by $\mathcal{N}_v$ the neighbors of a node v (not including v!). For a set of nodes $A \subseteq V$, $\mathcal{N}_A = \bigcup_{v \in A} \mathcal{N}_v$, i.e., the union of all neighborhoods of nodes in A. Have a look at the following (centralized) algorithm:

Input: a graph $G = (V, E)$
Output: a dominating set $S \subseteq V$

1: $S := \emptyset$.
2: For each $v \in V$ add a $w \in \mathcal{N}_v$ of maximum degree to S. If several neighbors have this degree, choose the one with smallest identifier.
3: $A := \{v \in V \text{ with } |S \cap \mathcal{N}_v| \geq 10\}$. $S := S \setminus \mathcal{N}_A$.
4: $S := S \cup A$.
5: $S := S \cup (V \setminus \mathcal{N}_S)$.
6: Return S.

---

a) (6) Describe verbally what the algorithm does!

b) (3) Prove that the algorithm indeed returns a dominating set!

c) (5) Explain how the algorithm can be translated into a synchronous distributed algorithm. How many rounds are at least required?

d) (4) Let M be a minimum dominating set of G. Prove that the nodes entering S in Line 2 that are not removed in Line 3 are at most $10|M|$ many.

e) (5) Show that the number of nodes chosen in Line 4 can be larger than $|M|$ by a factor of $\Omega(n)$, i.e., $|A| \in \Omega(n \cdot |M|)$!

f) (8) Assume that G is *regular*, i.e., all nodes have the same degree Δ. Prove that the worst-case approximation ratio of the algorithm is $\Theta(\Delta)$!