Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

FS 2016

Prof. Dr. Roger Wattenhofer
G. Bachmeier, P. Bissig, P. Brandes, S. Brandt,
C. Decker, P. Khanchandani, M. König,
D. Melnyk, L. Peer, Y. Wang

Exam
Principles of Distributed Computing

Friday, August 19, 2016
12:00 – 14:00

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

The exam lasts 120 minutes, and there is a total of 120 points. The maximal number of points for each question is indicated in parentheses. Your answers must be in English. Be sure to always justify your answers. Algorithms can be specified in high-level pseudocode or as a verbal description. 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. Please write legibly. If we cannot read your answers, we cannot grade them.

Please write down your name and Legi number (your student ID) in the following fields.

Name	Legi-Nr.

Points

Question Nr.	Achieved Points	Maximal Points
1		23
2		26
3		35
4		27
5		9
Total		120

1 Multiple Choice (23 Points)

Evaluate each of the following statements in terms of correctness. Indicate whether a statement is true or not by ticking the corresponding box. Each correct answer gets 1 point, **each wrong answer gets -1 point**. An unanswered statement gets 0 points. If the sum of collected points is negative, then you get 0 points for this question set.

Statement	**true**	**false**
It is possible to do uniform initialization of a wireless network of n nodes without collision detection in $O(n)$ time in expectation.	☐	☐
Uniform leader election in an n-node wireless network with collision detection is possible in $O(\log\log n)$ time w.h.p.	☐	☐
A minimum dominating set can be larger than a maximal independent set.	☐	☐
If a minimum dominating set is not an independent set, then nodes can be removed from it to get a maximal independent set.	☐	☐
For the function ONES : $\{0,1\}^k \times \{0,1\}^k \to \{0,1\}$ that evaluates to 1 if the two input k-bit strings contain the same number of ones and to 0 otherwise, it holds that $CC(\text{ONES}) \in \Omega(\log k)$.	☐	☐
If for a graph G it holds that $m \in \Omega(n^2)$ (where m is the number of edges and n the number of nodes of G), then the diameter of G can be computed in $O(\sqrt{n})$ time.	☐	☐
Given a weighted graph and its MST. If we add a single edge to the graph, a distributed algorithm can always update the MST in constant time.	☐	☐
Given a weighted graph and its MST. If we add a node and connect it with a single edge to any node previously in the graph, a distributed algorithm can always update the spanning tree in constant time.	☐	☐
In any anonymous graph in which we cannot deterministically find a leader, we also cannot deterministically construct a spanning tree.	☐	☐
Given an undirected graph with n nodes, if we need to query whether there exists a path between two nodes using a labeling scheme, the lower bound of the label size is $\Omega((\log n)^2)$.	☐	☐
Consider three graphs $G_1 = (V, E_1), G_2 = (V, E_2)$ and $G = (V, E)$ on the same node set, where E_1 and E_2 are two edge sets, and E is the union of E_1 and E_2. If there are adjacency labeling schemes of size k_i for G_i (for $i = 1, 2$), then there exists an adjacency labeling scheme of size $k_1 + k_2$ for G.	☐	☐
Synchronizer α may add a substantial overhead to the message complexity.	☐	☐
There exists some synchronous distributed algorithm that cannot be executed in an asynchronous environment without changing the asymptotic time complexity of the algorithm.	☐	☐

Statement	**true**	**false**
If a graph with n nodes can be colored with $c < n$ colors, then it can also be colored with $c+1$ colors.	□	□
When using Peterson's Algorithm deadlocks can occur.	□	□
Compare-and-Swap is considered more powerful than Test-and-Set.	□	□
When using the Arrow algorithm: If there is a quiescent moment, then all the arrows point towards the node holding the variable.	□	□
In the Arrow algorithm, the chosen spanning tree does not influence the runtime.	□	□
An advantage of the Ivy algorithm over the Arrow algorithm is that it also works in an asynchronous setting.	□	□
The self-stabilizing MIS (Algorithm 12.5) may fail to compute a locally stable solution if the adversary is allowed to modify the network topology.	□	□
In the Democrats and Republicans problem, if initially the majority of citizens votes for Democrats, there will eventually be no citizen who switches her opinion every other day.	□	□
If a distributed problem on an n-node graph can be solved in $O(n)$ time when the message size is restricted to $O(\log n)$ bits, then it can also be solved in $O(\log n)$ time when the message size is restricted to $O(n)$ bits.	□	□
There is a bipartite graph in which deterministic anonymous leader election is possible.	□	□

2 Low Cost Sorting (26 Points)

Consider a network of four nodes: v_1, v_2, v_3 and v_4. Each node is given an input number. An algorithm on this network executes in synchronous rounds. In each round, each node can participate in a compare-and-exchange operation with at most one neighbor. See Figure 1 for an example.

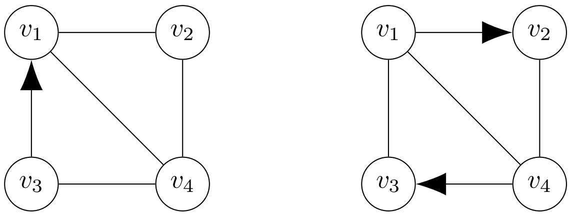

Figure 1: An example network where all pairs of nodes have edges between them except v_2 and v_3. After compare-and-exchange between a pair of nodes (represented by an arrow), the smaller value moves to the node at the head of the arrow, the larger to the node at the tail. In the first round, compare-and-exchange occurs between v_1 and v_3, the smaller value moving to node v_1. In the second round, compare-and-exchange occurs between v_2 and v_1, and between v_3 and v_4.

The cost of running an algorithm for t rounds on a network of m edges is $m \cdot t^2$. For questions **A)** and **B)**, give exact instead of asymptotic bounds.

A) Consider the problem of finding the minimum of the input numbers in a network with four nodes v_1, v_2, v_3 and v_4.

1) [2] Give a lower bound on the number of *edges* any network needs to put the minimum number at v_1.

2) [4] Give a lower bound on the number of *rounds* any algorithm needs to put the minimum number at v_1.

3) [3] Give a network and an algorithm for placing the minimum number at v_1 so that the cost $m \cdot t^2$ is minimum.

B) Now consider the problem of sorting the input numbers in a network with four nodes v_1, v_2, v_3 and v_4 so that node v_i gets the i^{th} smallest number.

1) [7] Give a lower bound on the number of *rounds* any algorithm needs to sort the numbers.

2) [5] Give a network and an algorithm to sort the numbers minimizing the cost $m \cdot t^2$ (better cost gives more points).

C) [5] Now you are given n nodes. Describe how to design a network and a sorting algorithm minimizing the cost $m \cdot t^2$. Please use the big-O notation for this part (better asymptotic cost gives more points).

3 1D Land Postal System (35 Points)

1D Land occupies a thin strip of land in the middle of the ocean. Recently, its population decided to implement a postal system. To do so the land is divided into n segments of equal length. For a packet to travel from one segment to a neighboring segment requires one time unit. Travel time within a segment is negligible.

To compete with modern technology such as email and delivery drones, a cannon is installed in each segment, which can safely and rapidly transport packets k segments in one direction within a single time unit. The cannon installations are fixed, i.e., cannons cannot turn or adjust their angle or power – each cannon always delivers to the same segment in distance k.

In the following we are interested in the MPTT (maximum packet travel time) which is the maximum over the shortest delivery times for every possible packet source and destination. All results and intermediate steps may be specified asymptotically using big O notation.

A) [5] Specify a k and cannon directions achieving a low MPTT. Lower MPTTs give more points.

Now suppose the cannon directions are chosen uniformly at random between *left* and *right*.

B) [3] For your algorithm (for your k): What are the worst possible cannon directions, and what is the MPTT in this scenario?

We are now interested in the PTTLWHP (packet travel time limit with high probability): a time limit in which a packet will reach its destination w.h.p., when sampling over the random cannon directions and packet source and destination.

C) [10] Show that the PTTLWHP is in $O(\frac{n}{k} \log n + k)$ for any k.

D) [10] Show a lower bound of $\Omega(\sqrt{n})$ for the PTTLWHP, for any k.

Hint: You may assume that if some predicate A does not hold in expectation, A also does not hold w.h.p..

You may now choose both the direction and the power of each cannon freely, i.e., you may choose any one target segment for every cannon.

E) [7] Specify a cannon configuration reaching a low MPTT (for any n). Lower MPTTs give more points.

4 Galactic Neighborhood (27 Points)

In the lecture, the concept of the neighborhood graph was used to prove a lower bound for coloring certain classes of graphs. We will look at another class of graphs here: stars. A star consists of a central node and a number of leaves (can be 0, 1, or more), each of which is connected with the central node, but no other nodes. Let $\mathcal{G}_k$ be the set of labeled stars with *up to and including* k nodes. We will look at the case of one round of communication where every node will learn exactly the labels of its immediate neighbors and nothing else, in particular not its neighbors' degrees.

A) [5] Draw the 1-neighborhood graph $\mathcal{N}_1(\mathcal{G}_3)$ for labeled stars with up to 3 nodes where each star can use labels from $\{1, 2, 3\}$.

B) [5] Give a deterministic distributed algorithm that colors a star with k nodes where $k \leq 3$ with labels from $\{1, 2, 3\}$ with two colors in 1 round, or prove that such an algorithm does not exist.

Now we restrict the labels that can occur in a star with k nodes to $\{1, \ldots, k\}$, i.e. every label from 1 to k occurs exactly once.

C) [5] Draw the 1-neighborhood graph $\mathcal{N}_1(\mathcal{G}_3)$ for labeled stars with up to 3 nodes where each star with k nodes uses exactly the labels $\{1, \ldots, k\}$.

D) [5] Give a deterministic distributed algorithm that colors a star with $1 \leq k \leq 3$ nodes with labels from $\{1, \ldots, k\}$ with two colors in 1 round, or prove that such an algorithm does not exist.

E) [7] Prove or disprove that the result of D) generalizes to stars with k nodes and labels $\{1, \ldots, k\}$ for arbitrarily large k.

5 Reading Assignment (9 Points)

A) [3] Show that there is an algorithm for online graph exploration that is 1-competitive on weighted trees.

B) [3] Does the hDFS algorithm have a constant competitive ratio on weighted paths?

C) [3] Is there an algorithm for online graph exploration that has a competitive ratio of strictly less than 16 on planar graphs with unit-weight edges?