

Instructions. There are five questions, and you must answer *at least four questions* in order to pass the exam.

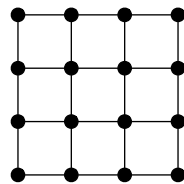
You are free to look at any source material (this includes lecture notes, textbooks, and anything you can find with Google), but you are not allowed to collaborate with anyone else or ask for anyone's help (this includes collaboration with other students and asking for help in online forums).

You are free to use any results from the lecture notes directly (including the results from the regular exercises). You are free to use computers and computer programs to find solutions.

If you cannot solve a problem entirely, please try to at least explain what you tried and what went wrong.

* * *

Question 1: Maximal matching. Consider the following graph G_1 with 16 nodes:



Design a deterministic PN-algorithm A that finds a *maximal matching* in graph G_1 . That is, if you run A in any simple port-numbered network N that has graph G_1 as the underlying graph, algorithm A will stop and output a maximal matching.

The local inputs of the nodes are empty. Any running time is fine. A brief, informal description of the algorithm is sufficient. The algorithm does not need to do anything meaningful if you run it in any other graph.

Question 2: Edge coloring. Let graph G_1 be as defined in question 1. Prove that there is no deterministic PN-algorithm that solves the following problem: finding a *proper edge coloring* (with any number of colors) in graph G_1 .

Hint: You are expected to use an argument based on covering maps.

There is more, please see the other page...

Question 3: Two colors. Design a deterministic PN-algorithm that solves the following problem in $O(1)$ rounds:

- Graph family: bipartite graphs.
- Local inputs: proper vertex coloring with 2 colors.
- Local outputs: label the edges with labels R, G, and B such that for any node v of degree at least 10, the edges incident to v use **at least two different labels**.

A brief, informal description of the algorithm is sufficient.

Question 4: Three colors. Prove that there is no deterministic PN-algorithm that solves the following problem in $O(1)$ rounds:

- Graph family: bipartite graphs.
- Local inputs: proper vertex coloring with 2 colors.
- Local outputs: label the edges with labels R, G, and B such that for any node v of degree at least 10, the edges incident to v use **all three labels**.

Hints: Consider $(10, 10)$ -biregular trees. Compare with the sinkless orientation problem.

Question 5: One red, one green. Consider the following problem in $(10, 10)$ -biregular trees: label the edges with labels R, G, and B such that for any node v of degree 10, the edges incident to v use label R **exactly once** and label G **exactly once**.

- (a) Formalize this as a *bipartite locally verifiable problem* Π (Section 9.1.1 in the lecture notes).
- (b) Show that Π is not solvable in 0 rounds.
- (c) Show that Π is a periodic point in round elimination, i.e., if you apply re sufficiently many times, you will arrive at a problem equivalent to Π .
- (d) How long is the period, i.e., how many applications of re are needed to arrive at the original problem Π ?