COMP 182: Algorithmic Thinking
23 January 2014

A k-coloring of an undirected graph $g = (V, E)$ is an assignment of integers $\{1, 2, \ldots, k\}$ (the colors) to the nodes of g such that no two neighboring nodes have the same color. A graph is k-*colorable* if it has a k-coloring. The *chromatic number* of graph g is the smallest k such that g is k-*colorable*.

1 Do you understand the definitions?

1. Give an example of a graph with at least 5 nodes that is 3-*colorable*.
2. Give an example of a graph with at least 5 nodes that is not 3-*colorable*.
3. Give an example of a graph whose *chromatic number* is 4.

2 Problem formulations

Define formally (1) the problem of checking whether a graph is k-*colorable* and (2) the problem of computing the *chromatic number* of a graph.

3 Algorithms and their efficiency

1. Give the pseudo-code of two algorithms **IsKColorable** and **ComputeChromaticNumber** to solve the two problems you defined. Your **ComputeChromaticNumber** must make use of your **IsKColorable** algorithm (that is, by calling it, and not by duplicating it).
2. What is the input size to each of your algorithms.
3. What is the number of steps that each of the two algorithms take.

4 Problem reduction

For each of the following problems, formulate it as a graph coloring problem.

1. Assume an airline company wants to schedule n flights in one day, where flight f_i ($1 \leq i \leq n$) is during the time interval (a_i, b_i). The airline company is interested in using the fewest number of airplanes to schedule all flights (an airplane cannot be used for two flights at the same time).
2. We want to fill a 9×9 sudoku puzzle that is partially completed (you need to modify the problems above slightly).