

CS5234 : Combinatorial and Graph Algorithms Homework Set #6

Out: 18 Oct 2003 Due: 1 Nov 2003

1. On-line algorithm for convex hull computation in linear time

You receive a stream of points $p_1(x_1, y_1)$, $p_2(x_2, y_2)$, ... with the property that the x -coordinates are monotonically increasing: $x_1 < x_2 < \dots$. At any time step k , you must compute the upper half of the convex hull of the points p_1, \dots, p_k . Design and program an animated algorithm to do this.

Show that:

- a) extending the convex hull by adding a single point p_k may require time $\theta(k)$,
- b) computing the convex hull of all k points p_1, \dots, p_k can be done in time $O(k)$.

2. Comment on the following "proofs":

Thm: All cats are of the same color.

Proof by induction on the number of cats.

Base: consider any set S_1 consisting of a single cat. Obviously, all cats in S_1 have the same color.

Induction step: Assume the theorem is correct for all sets S_k consisting of k cats, and consider an arbitrary set S_{k+1} consisting of $k+1$ cats. Remove cat C' from S_{k+1} , resulting in a set S'_k of k cats, all of the same color. Put C' back in and remove another cat C'' from S_{k+1} , resulting in a set S''_k of k cats, again all of the same color. Thus, all cats in S_{k+1} have the same color, QED.

Thm: All natural numbers (positive integers) are interesting.

Proof by contradiction: Assuming there are uninteresting natural numbers, it follows that there is a least uninteresting natural number. But the property of being the least uninteresting number clearly makes that number interesting, thus contradicting the assumption, QED.

3. Tiling chess boards

- a) Consider an $n \times n$ grid, n even, and imagine its squares painted black and white alternatingly, as on a chess board. We aim to cover (tile) this board with dominoes, i.e. 2×1 rectangles that cover two adjacent squares. Prove or disprove the following statements:

a1) If we cut off two diagonally opposite corner squares, the mutilated chess board can be tiled with dominoes

a2) If we cut out any one white square and any one black square, the mutilated chess board can be tiled with dominoes.

- b) Consider an $n \times n$ grid, n a power of 2.

Prove or disprove:

If we cut out any single square, the mutilated chess board can be tiled with L-shaped triominoes, i.e. 3 adjacent squares in the shape of an L.

4. Program the Kara system for "Island Hopping"

Download Kara: www.educeth.ch/compscience/karatojava/kara/index.html

In the grid world of Kara there are "islands", each one represented by a single tree trunk. We are interested in finite state machines (programs) that send Kara on a voyage to visit many islands, under some assumptions on the relative location of the islands in the ocean. In order to state this problem precisely we introduce some standard mathematical concepts.

The *Manhattan distance* between two grid squares (x, y) and (x', y') is defined as $|x - x'| + |y - y'|$. We assume that the distance between any pair of islands is ≥ 3 .

Two islands (x, y) and (x', y') are called *neighbors* iff their relative location is like that of a knight's jump in chess, i.e. $|x - x'| = 2$ and $|y - y'| = 1$, or $|x - x'| = 1$ and $|y - y'| = 2$.

A maximal set of islands such that one can go from any island to any other in the same set is called an *archipelago*. An archipelago can be represented as a connected graph whose nodes are islands and whose edges are pairs of neighbors.

A sequence of consecutive edges that leads back to the starting node is called a cycle. A graph without cycles is called a tree.

- a) Program Kara so that, when started on a square next to an island, he will visit (touch) all the islands of any archipelago whose graph is a tree. If the graph has cycles, what can you say about the part of the archipelago your program will visit?
- b) Modify the program a) so that Kara lays down a trace of clover leaves covering every square he has traversed, and stops when he returns to the starting square.
- c) Having learned to explore an archipelago, Kara aims to explore the entire ocean of his toroidal world (the world is a torus: when Kara gets beyond a border of the grid, he automatically reappears on the opposite side). For this challenging task, Kara is allowed to place and pick up clover leaves to mark and unmark any square. The general idea is that, when Kara has finished visiting one archipelago, he sails out into the open seas, hoping to discover new archipelagos he has not visited before. If Kara is deterministic, can you guarantee that Kara eventually discovers every archipelago? Also, experiment with the non-deterministic version of Kara.