Master of Science in Informatics at Grenoble

UE Mathematics for Computer Science

Final exam December 14, 2009 (3 hours)
Neither documents nor calculators are allowed. Only personal hand-written notes are allowed. Use separated sheets for problem 1,2 and problem 3. All problems are independent from each other.

Problem 1: Paths in directed graphs *(6 points)*

Question 1.1 : Degrees

Prove that in all directed acyclic graphs there exists at least one vertex which has no predecessor (i.e. its in-degree is equal to 0).

Let $G = (X, E)$ a graph with the adjacency integer matrix A. Denote by $a_{i,j}^{(n)}$ the number of length n paths from i to j.
Question 1.2 : Number of length n paths

Prove that $a_{i,j}^{(n)}$ is the coefficient (i, j) of the matrix A^n.

Question 1.3 : Directed acyclic graphs

When the graph G is acyclic, prove that there is some k_0 for which $A^k = 0$ for all $k \geqslant k_0$. Deduce that the total number of paths $c_{i,j}$ from i to j is finite.

Question 1.4 : Number of paths

Prove that $c_{i,j}$ is the coefficient (i, j) of the matrix $(Id - A)^{-1}$.

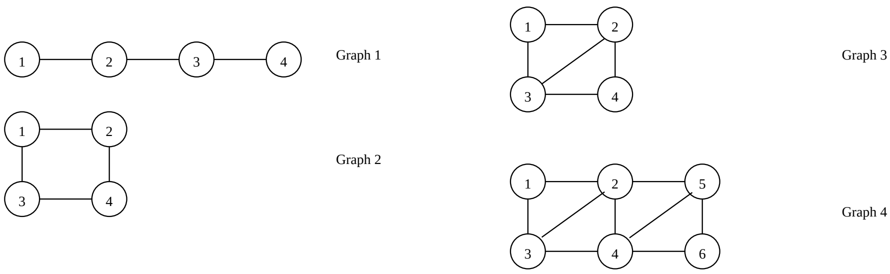
Problem 2: undirected graphs, degrees and random walks *(8 points)*

Question 2.1 : Vertices

Prove that every finite undirected graph with two or more vertices has two vertices of the same degree.

The *Page Rank* is commonly used by search engines to classify web pages according to their ”popularity”. The concept developed by this algorithm consists in a random walk on web pages that computes automatically the frequency of visits to web pages. Then the answers to search a request is a set of pages ranked according the frequency. The implementation of page rank is realized via a software robot that jumps from pages to pages in the web. When the robot is on one page the next jump is chosen uniformly among the links on the current page.

This algorithm is modelled by a random walk on the graph of web pages. The page pointed by the robot at time n is a homogeneous Markov chain. Moreover we suppose that in one jump the robot could stay on the node, go back to the previous page or jump to another page poitned by the current one and that all these events have the same probability. (We consider the transition graph as symmetric, with a loop on each node). The ”popularity” corresponds to the steady-state probability of this Markov chain.



Question 2.2 :

Under which condition the popularity is uniquely defined ? Is this condition satisfied in the current Web ?

Question 2.3 :

For each graph in the figure, write the transition matrix and compute the steady state probability of each page.

Question 2.4 :

Analyse your results and propose a general formula for a graph with nodes having all the same degree.

Question 2.5 :

Generalize to an arbitrary graph.

Question 2.6 :

Explain why the situation changes when the graph is directed.

Problem 3: Recursion – analysis of the Towers of Hanoi *(8 points)*

The problem is to solve an old puzzle that comes from a vietnamese legend. The original form consists in 3 rods and n disks of different diameters which can slide onto any rod. The process starts with the disks in a neat stack in ascending order of size on one rod, the smallest being at the top. The objective is to move the entire pile to another rod, obeying the three following rules: only one disk can move at a time, each move consists in taking the upper disk from one of the rod and place it on the top of the other disks that may already be present on another rod, and finally, no disk may be placed on top of a smaller one.

In the following $H(n, 3)$ will denote the number of steps for moving the entire pile of n disks from the first rod to the third one. Let extend this notation to $H(n, k)$ when there are k rods. There exist many variations of the problem, the goal of this problem is to investigate some of these variants from the mathematical point of view.

Question 3.1 : Upper bound

Determine the number of all the possible moves.

Question 3.2 : Standard recursive solution

The solution of the basic problem is well-known using the decomposition of Hanoi(n) from peg 1 to 3 using peg 2 as an intermediate peg. if $n \neq 0$ then, we do:

- Hanoi(n-1) from peg 1 to peg 2 using the intermediate peg 3
- move the remaining disk from 1 to 3
- Hanoi(n-1) from peg 2 to peg 3 using the intermediate peg 1

We are looking for the total number of moves.

Give the expression of $H(n)$ using $H(n - 1)$ and deduce its value by a closed formula in function of n.

Question 3.3 : Coding the standard recursive solution

The disk positions and the successive moves may be determined using a binary representation. A first way is to associate a binary digit to each disk where the most significant (leftmost) bit represents the largest disk (0 indicates that it is on the intial peg and 1 that it has been moved to its destination). Then, the bit-string is read from left to right and each bit can be used to determine the location of the corresponding bit.

Explain how to realize such a coding and give the corresponding sequence for Hanoi(4).

Question 3.4 : Improving the coding

The *Gray code* gives an alternative way for solving the problem. In this coding, the numbers are expressed in a binary combination but the transitions operate on the premise that each value differs by its predecessor by only one bit changed. The idea applied to the tower of Hanoi is to code the moves of the disks. We start at 0 and counts up. The bit changed at each move corresponds to the disk to move, where the least significant bit is the smallest disk and the most significant one is the largest. Here, we identify only which disk to move, but not where to move it.

Give the sequence of moves for Hanoi(4) using the Gray code.

Question 3.5 : Increasing the number of pegs

In this part, we assume that there are $n+1$ pegs (the origin, the destination and $n-1$ intermediate pegs).

Construct a feasible solution in this case and compute its number of moves $H(n, n+1)$.

Question 3.6 : Determining a "good" number of pegs

Assume now that n is a perfect square and there are $\sqrt{n}+1$ pegs (the origin, the destination and the intermediate pegs). The idea is to split the pile of disks in series of $\sqrt{n}$ consecutive disks and to move them by packets.

Give the sketch of the corresponding solution and computes its cost $H(n, \sqrt{n}+1)$.