Master of Science in Informatics at Grenoble

UE Mathematics for Computer Science

Final exam December 14, 2011 (3 hours)
Only personal hand-written notes are allowed.
Use separated sheets for problems 1-2 (part I) and problems 3-4 (part II).
All problems are independent from each other.
Number of points given for each problem is given for information purposes only and is subject to modifications without notice.

Part I

Problem 1: More about primes... *(4 points)*

Question 1.1 : A first (and old) result

Euclide proved that there are an infinity of primes. The principle is to consider any finite set of primes: $p_1, p_2, ..., p_n$ and to consider the integer $q = p_1 p_2, ...p_n + 1$.
First, what happens if this new number is not a prime?
If it is, prove first that it does not belong to the initial set.
Deduce that there are an infinite number of primes.

Question 1.2 : There are an infinite number of primes...

Euler proved in the XIX-th century the same result using another argument.
The principle is as follows: Let $\mathcal{P}$ denote the set of primes, and let us consider the product $\Pi_{p\in\mathcal{P}} \frac{1}{1-1/p}$.

1. Rewrite this product using a geometric series (you should explicit the generic term)
2. Show by using the fundamental theorem of arithmetic that:

$$\Sigma_{k\geqslant 0} \frac{1}{\Pi p^k} = \Sigma_n \frac{1}{n}$$

and then, deduce that $\mathcal{P}$ is infinite.

Problem 2: Bin Packing... *(5 points)*

Given a set of n items, each one of height s_i (for $1 \leqslant i \leqslant n$) and an integer H, the problem is to determine the minimum number of bins needed to store all the items. The constraint here is that the total heights of the items selected to fit into a bin is less than H and of course, $\forall i, s_i \leqslant H$. In all the following, we will denote by N_A the number of bins obtained by applying algorithm A. N_{opt} is the optimal number of bins.

Question 2.1 :

Show briefly that this problem can be formulated as a problem where the size of the bins are unitary with rational items. Then, propose a solution for the following instance:
$n = 10$ and $(\frac{1}{9}, \frac{1}{3}, \frac{5}{9}, \frac{4}{9}, \frac{1}{9}, \frac{1}{3}, \frac{1}{9}, \frac{2}{9}, \frac{1}{3}, \frac{4}{9})$.

The general problem is NP-hard. Let us study the mathematical properties on two particular cases.

Question 2.2 : Small items

We consider here the following assumption: the maximal height of an item is limited to ρ (where $0 < \rho \leqslant 1$). Let analyze the following process (called FF – First Fit): The items are first sorted by decreasing heights, then, the items are filled one after the other in the first available bin where it fits.

Prove the following properties.

1. **Case 1** $\rho \leqslant \frac{1}{2}$. Show that $N_{FF} \leqslant (1 + 2\rho)N_{opt} + 1$[1].
2. **Case 2** $\rho \geqslant \frac{1}{2}$. Using a lower bound argument, show that $N_{FF} \leqslant 2.N_{opt} + 1$. Deduce the following bound $N_{FF} \leqslant (1 + 2\rho)N_{opt} + 1$.

Question 2.3 : Limited number of items

We assume in this question that there are a maximum of k different sizes (that means several items have the same size). The idea here is to gather the items in k groups. Let us denote by G_i the corresponding groups ($1 \leqslant i \leqslant k$). Finally, we denote by m the maximum number of items of the same size.

1. Explicit the groups and all the parameters on the example of question 2.1.

In the general case, the number of all sub-sets of n elements is exponential in n. Here, there is no special order on each group and the number of groups k is fixed.

2. Show that the number of all possible configurations is polynomial[2]
3. How difficult the problem is within each group? What could be the interest of such a result in the perspective of designing an algorithm in the general case.

[1] You can use the following identity (after given its justification: $\frac{1}{1-x} \leqslant 1 + 2x$ for $\frac{1}{2} \leqslant x < 1$)

[2] One way may be to enumerate all the configurations by a vector of k elements.

Part II

Problem 3: Pattern recognition *(5 points)*

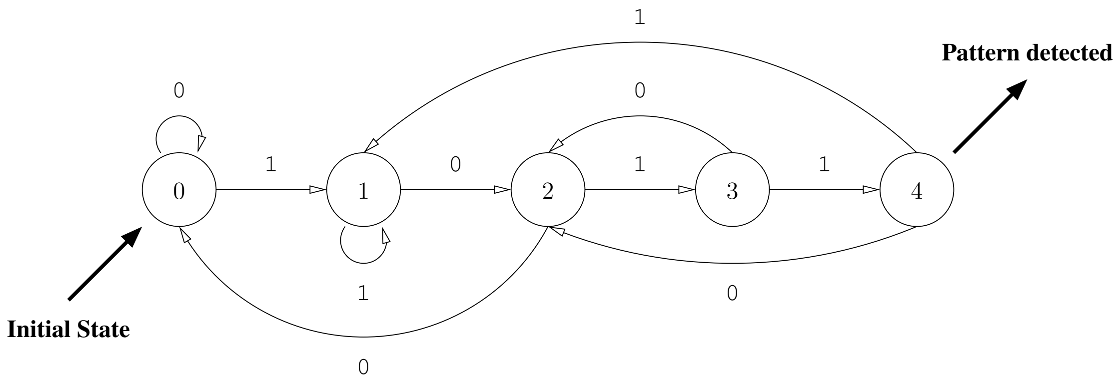
An infinite sequence S of bits is analyzed on-line to detect a specific pattern P. For this pattern, a small automaton is built and the detection (a trigger) occurs at time n when the pattern P ends at the n^{th} bit.

In the following example, with the pattern P=[1011] the automata triggers at time 9 and 12

$Time$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Bit	0	0	0	0	1	0	1	0	1	1	0	1	1	0	0	1

Question 3.1 : Automata

Show rapidly that the following automaton detects pattern [1011].



Suppose now that the sequence S of bits is random and generated independently and uniformly on $\{0, 1\}$.

Question 3.2 : Average time to the first pattern

For such a uniform sequence S give the expected time of the first occurrence of the pattern [1011].

Question 3.3 : On the long-run

For the infinite sequence S, compute the frequency of triggers.

Question 3.4 : Generalization

Are the time to the first pattern and the frequency depending on the pattern itself ?

Give a counter example if necessary.

Problem 4: On hats at a party *(6 points)*

"At a party, suppose n men give their n hats to a hat-check person. Let d_n be the number of ways that the hats can be given back to the men, each man receiving one hat, so that no man receives his own hat. "[3]

Question 4.1 : Small values of n

Give all the possibilities for $n = 1, 2, 3, 4$, deduce the corresponding values of d_1, d_2, d_3, d_4.

Question 4.2 : Recurrence equation for d_n (2 steps)

Establish the following recurrence equation by a combinatorial argument

$$d_{n+1} = n(d_n + d_{n-1}) \tag{1}$$

Question 4.3 : Recurrence equation (1 steps)

Deduce that

$$d_n = nd_{n-1} + (-1)^n,$$

and the formula

$$d_n = n! \left(\frac{1}{0!} - \frac{1}{1!} + \frac{1}{2!} - \cdots + \frac{(-1)^n}{n!} \right).$$

Question 4.4 : Random hat-check person

Suppose the hat-checker a completely random-uniform person.

1. What is the probability p_n that no man receives his own hat ?
2. Compute the asymptotic value p_∞ of p_n and compute the order of magnitude of the approximation error.
3. Give p_{13} with a precision of 10^{-2}.

Question 4.5 : Random hat-check person returns

Compute the average number of men receiving their own hat. *Hint: Consider the random variable X_i indicating if man i gets his hat or not*)

Question 4.6 : A computer hat swapper

For this question, we have a `Random()` generator that provides independent samples of uniformly distributed on $[0, 1)$ random variables.

1. Write a simulation algorithm that generates uniformly a permutation of hats.
2. Write a simulation algorithm based on the rejection method that generates uniformly a permutation without fixed points and compute its average cost.
3. From Equation (1), write a uniform generator of a permutation without fixed point.

[3] From *Enumerative Combinatorics*, R. Stanley 2011