Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
Hamid Ghodselahi October 31, 2012

Algorithm Theory, Winter Term 2012/13
Problem Set 2

hand in by Wednesday, November 14, 2012

Exercise 1: Polynomial Multiplication using FFT [8 Points]

Compute the coefficient representation of $(p(x))^2$ by using the FFT algorithm.

$$p(x) = 2x^3 - x^2 + 4x + 1$$

Exercise 2: Distribution of Sum [6 Points]

Let A and B be two sets of integers between 0 and $n-1$, i.e., $A, B \subseteq \{0, \ldots, n-1\}$. We define two random variables X and Y, where X is obtained by choosing a number uniformly at random from A and Y is obtained by choosing a number uniformly at random from B. We further define the random variable $Z = X + Y$. Note that the random variable Z can take values from the range $\{0, \ldots, 2n-2\}$.

(a) Give a simple $O(n^2)$ algorithm to compute the distribution of Z. Hence, the algorithm should compute the probability $\Pr(Z = z)$ for all $z \in \{0, \ldots, 2n-2\}$.

(b) Can you get a more efficient algorithm to compute the distribution of Z? You can use algorithms discussed in the lecture as a black box. What is the time complexity of your algorithm?

Hint: Try to represent A and B using polynomials.

Remark: Exercise 2 was an exam question in fall 2012.

Exercise 3: Extended Interval Scheduling [8 Points]

A generalized version of interval scheduling problem can be defined as follows:

- **Given** are a set of intervals $[a, b]$ such as in the original interval scheduling problem.
- **Goal** Select a largest possible subset of the intervals such that at no time more than k intervals overlap. As before, overlaps just at the boundary don't count, e.g., $[1, 2]$ and $[2, 5]$ are not overlapping at time 2.

(a) Find an optimal greedy algorithm for the case $k = 2$ and show that your algorithm computes an optimal solution.

Hint: The algorithm from the lecture solves the case $k = 1$.

(b) Describe an efficient implementation of your algorithm and give the running time of your implementation.

Exercise 4: Matroids [8 Points]

We have defined matroids in the lecture. For a matroid (E, I), a maximal independent set $S \in I$ is an independent set that cannot be extended. Thus, for every element $e \in E \setminus S$, the set $S \cup \{e\} \notin I$.

a) Show that all maximal independent sets of a matroid (E, I) have the same size. (This size is called the rank of a matroid.)

b) Consider the following greedy algorithm: The algorithm starts with an empty independent set $S = \emptyset$. Then, in each step the algorithm extends S by the minimum weight element $e \in E \setminus S$ such that $S \cup \{e\} \in I$, until S is a maximal independent set. Show that the algorithm computes a maximal independent set of minimum weight.

c) For a graph $G = (V, E)$, a subset $F \subseteq E$ of the edges is called a forest iff (if and only if) it does not contain a cycle. Let $\mathcal{F}$ be the set of all forests of G. Show that $(E, \mathcal{F})$ is a matroid. What are the maximal independent sets of this matroid?