Randomized Algorithms and Probabilistic Methods

**General Information:** In the first two weeks of the term we will hand out an exercise sheet on Tuesday. Some hints to get you started will be given in the exercise session that day. The solutions will be discussed in the exercise class of the following week.

You are encouraged to hand in your solutions to any of the assistants for feedback.

**Exercise 1**

We play a game with a randomly shuffled deck of $2n$ cards, $n$ black and $n$ red. The game works as follows, I reveal the cards one by one but before I reveal a card you can decide if you want the game to end with that card or not (this decision must be made before we run out of cards in the deck). If the card revealed when the game ends is red you win, otherwise me.

What is the optimum strategy in this game and depending on $n$, how much better can you do than just reveal the top card?

**Exercise 2**

Suppose that you are given a Las Vegas algorithm which has running time $t(|I|)$ when executed on instance $I$. Here $|I|$ denotes the size of the instance, i.e., the number of bits that are necessary to encode it. Design a (randomized) algorithm that always calculates a correct solution, and determine its expected running time!

**Exercise 3**

We start with a bag containing exactly one red and one blue ball. In each step we pick one of the balls in the bag uniformly at random, clone that ball, and put both balls (the original and the cloned one) back into the bag. Hence, after $n$ steps there are $n + 2$ balls in the bag some of which are blue and some of which are red.

For any integers $n \geq 0$ and $1 \leq k \leq n + 1$, what is the probability to have exactly $k$ blue balls in the bag after $n$ steps?

**Exercise 4**

We roll a six-sided fair dice over and over. What is the expected number of throws until we see two consecutive sixes for the first time?
Exercise 5

Let $M$ be a set. An $r$-coloring of $M$ is a mapping $c : M \to \{1, \ldots, r\}$. We say that a subset $X \subseteq M$ is monochromatic if all elements contained in $X$ are colored with the same color, i.e., if for all $x, y \in X$ we have $c(x) = c(y)$.

Suppose now that you are given a family $F = \{F_1, \ldots, F_s\}$ with $|F_i| = t$ and $F_i \subseteq M$ for $1 \leq i \leq s$. Prove that if $s < 2^{t-1}$, there exists a 2-coloring $c$ such that no set $F_i$ is monochromatic! Can you give a similar bound for $r$-colorings, $r \geq 3$?

Exercise 6

Let $m, n$ be two positive integers and let $f : \{0, \ldots, n-1\} \to \{0, \ldots, m-1\}$ be a function such that for all $x, y \in \{0, \ldots, n-1\}$ we have $f((x + y) \mod n) = (f(x) + f(y)) \mod m$. The only way we have for evaluating $f$ is to use a lookup table that stores the values of $f$. Unfortunately, an evil adversary has corrupted up to $\frac{1}{5}$ of the table entries.

Provide a simple randomized algorithm that for every input $x \in \{0, \ldots, n-1\}$ outputs a value $y \in \{0, \ldots, m-1\}$ that equals $f(x)$ with probability at least $1/2$. Try to use as few lookups as possible.

Can you improve on the error probability by using more lookups?

Discussion of the homework on September 22nd, 2015.