Institut für Informatik
Prof. Dr. Heiko Röglin
Dr. Melanie Schmidt

universitätbonn

Randomized Algorithms and
Probabilistic Analysis
Summer 2017

Problem Set 3

Please hand in your solutions in Monday's lecture on the 15th of May (or via e-mail).

Problem 1

1. A lively monkey types $26^6 \cdot 42 + 5$ letters (=499017796 letters) on a keyboard. We assume that the keyboard has only upper-case letters and that each of the 26 letter is chosen uniformly at random. What is the expected number of times that the word `RANDOM` appears?

2. We flip a fair coin $n + \log_2 n$ times, assume that n is a power of two. We get a sequence $x_1, x_2, \ldots, x_{n+\log_2 n}$ with $x_i \in \{H, T\}$. We say that $x_i, \ldots, x_{i+\ell-1}$ is an *ℓ-sequence* if $x_i = x_{i+1} = \ldots = x_{i+\ell-1}$ (all heads or all tails). What is the expected number of ℓ-sequences for $\ell = 1 + \log_2 n$?

Problem 2

Give an example of two random variables $X, Y : \Omega \to \mathbb{N}$ where $E(X), E(Y)$ and $E(X \cdot Y)$ exist, but $E(X \cdot Y) \neq E(X) \cdot E(Y)$. Bonus task: Give an easy example of two random variables $X, Y : \Omega \to \mathbb{R}$ where $E(X)$ and $E(Y)$ exist, but $E(X \cdot Y)$ does not exist.

Problem 3

(Recall that $\sum_{i=1}^{n} \frac{1}{i} = H_n = \Theta(\log n)$)

1. Assume that we have n images, and that n is a multiple of k. Each image shows a portrait of a person, and there are k different persons. There are n/k images of each person. We want to have an (arbitrary) collection with exactly one picture of each person and use the following randomized algorithm: We keep choosing a picture uniformly at random (with replacement) until we have pictures of k different people. If we already have a picture of a person, we discard the chosen picture, otherwise we add it to our picture collection. What is the expected numer of times that we choose a picture until we have our collection of k pictures of different persons?

2. We want to sort n distinct numbers that are stored in array A. We use *GuessSort*: We pick two indices $i, j \in \{1, \ldots, n\}$ uniformly at random from all possible pairs (i, j) with $i < j$. If $A[i] > A[j]$, we swap the elements, otherwise, we do nothing. What is the expected number of comparisons that this algorithm does until the array is sorted?

Problem 4

We are given a data stream of numbers $a_1, a_2, a_3, \ldots$ (of unknown length) and want to sample one number s. Instead of ReservoirSampling, we use the following algorithm: Intially, we store a_1 in s. Then, after each a_i, we keep the current s with probability $1/2$ and replace it with a_i with the remaining probability. What is the probability $\Pr(s = a_j)$ for $j \in \{1, \ldots, j\}$ after processing a_i?