Lecture

At the lecture we continue with randomized algorithms. We will talk about random variables, hash tables, and hash functions.

1 [w] Expected values

1.1 Let X be a random variable which assumes the values 2, 5 and 8 with probabilities 1/3, 1/2 and 1/6 respectively, i.e., $P[X = 2] = 1/3$ etc. What is the expected value of X?

1.2 Let X be a random variable which assumes the values 2^i, ..., for $i = 0, 1, 2, \ldots, k$ with probabilities $2^{-(i+1)}$ respectively, i.e., $P[X = 2^i] = 1/2^{(i+1)}$ etc. What is the expected value X?

2 [w] Run by Hand and Properties

2.1 Insert the key sequence $K = 7, 18, 2, 3, 14, 25, 1, 11, 12, 1332$ into a hash table of size 11 using chained hashing with hash function $h(k) = k mod 11$.

2.2 Let K be a sequence of keys stored in a hash table A using chained hashing. Given A, can one efficiently find the maximum element in K?

3 Streaming Statistics An IT-security friend of yours wants a high-speed algorithm to count the number of distinct incoming IP-addresses in his router to help detect denial of service attacks. Can you help him?

4 Multi-Set Hashing A multi-set is a set M, where each element may occur multiple times. Design an efficient data structure supporting the following operations:

- ADD(x): Add an(other) occurrence of x to M.
- REMOVE(x): Remove an occurrence of x from M. If x does not occur in M do nothing.
- REPORT(x): Return the number of occurrences of x.

5 Christmas party at DTU (exam 2015) During the Christmas party at DTU party the Dean suddenly wants to know who won most Christmas cookies in the "Bing or Ding" game. He suggests the following algorithm:

Algorithm 1: Find student with most cookies

```
max ← −∞
s ← null
Randomly order the students. Let s_1, ..., s_n be the students in this random order.
Let c_i denote the number of cookies won by student s_i.
for i = 1, ..., n do
  | if if c_i > max then
  | | max ← c_i and s ← s_i    (*)
  | end
end
return s
```

In the following assume that all students won a different amount of cookies. That is, $c_i \neq c_j$ for all $i \neq j$.

5.1 What is the probability that the line (*) is executed at the last iteration?

5.2 Let X_i be a random variable that is 1 if line (*) is executed in iteration i and 0 otherwise. What is the probability that $X_i = 1$?

5.3 What is the expected number of times line (*) is executed?

6 Dynamic hashtable Explain how to make a dynamic hashtable with insertions using the doubling technique. Your solution should use $\Theta(n)$ space, where n is the number of elements in the hashtable. What is the insertion time of your solution?

7 Boxes of beer You are given n boxes $B_1, \ldots, B_n$. Exactly k boxes contain a bottle of beer ($k \leq n$) the rest is empty. From the outside one cannot see whether a box is empty or not. The aim is to find a box with a beer it. The following deterministic algorithm is suggested: Open the boxes $B_1, B_2, \ldots$ in this order. The algorithm stops when a beer is found. We count opening a box as one computational step.

7.1 What is the best-case running time of the deterministic algorithm.

7.2 What is the worst-case running time of the deterministic algorithm.

7.3 Consider the following randomized algorithm: Randomly pick a number $i \in \{1, 2, \ldots, n\}$ and open box B_i.

1. What is the expected running time of this algorithm?
2. What is the worst-case running time of this algorithm?

7.4 Now consider a modification of the randomized algorithm above: Randomly pick a box you have not opened before and open it.

What is the worst case running time of this algorithm?

We now want to analyse the expected running time of this algorithm. Make a indicator variable X_i for each empty box i, where $X_i = 1$ if box i was opened by the algorithm and 0 otherwise. Let X be the number of boxes opened by the algorithm.

1. Express X using the X_is, i.e., $X =$ ......
2. What is the expected value of X_i?
3. What is the expected value of X (use the bounds on the expected value of the X_is)?
4. What is the expected running time of the algorithm?