Assignment 2

Advanced Algorithms and Datastructures

Authors:
Jenny-Margrethe Vej (rwj935)
Martin Gielsgaard Grünbaum (wrk272)
Martin Nicklas Jørgensen (tzk173)

May 22, 2014

1 Hash functions for sampling

1.1 Exercise 1(a)

1.2 Exercise 1(b)

2 Bottom-k sampling

2.1 Exercise 2

2.2 Exercise 3(a)

We would store the buttom-k samples in a minimum heap structure H, sorted by their hashing value. This way we can insert new entries in $O(\lg\ n)$, and retrieve the $S_h^k(H)$ lowest hash values in $O(k\ \lg\ n)$ where n is the total number of input values.

2.3 Exercise 3(b)

As written above we would be able to process/insert the next key in $O(\lg\ n)$ time.

2.4 Exercise 4

2.5 Exercise 4(a)

We will prove the equality $S_h^k(\mathrm{A} \cup \mathrm{B}) = S_h^k(S_h^k(A) \cup S_h^k(B))$. We can see each set as a sorted stack that keeps the smallest values at the top. The left hand part of the equality $(S_h^k(\mathrm{A} \cup \mathrm{B})))$ corresponds to merging the two stacks and taking the k top values. The right hand side $(S_h^k(S_h^k(A) \cup S_h^k(B)))$ corresponds to taking the k topmost values from both stacks and then merging them and taking the k smalles values from the resulting stack.

Since we take the k smallest values from each stack we are guaranteed to have the smallest value from the union of A and B, thus proving the equality.

2.6 Exercise 4(b)

2.7 Exercise 4(c)

3 Bottom-k sampling with strong universality

3.1 Exercise 5

3.2 Exercise 6

3.3 Exercise 7

References