Exercise 9.4

Monday, 9 October 2023 14:22

Input:

either need $a_0, \ldots, a_{m-1}$

A stream $\langle a_1, \ldots, a_m \rangle$ of m distinct items in the vanilla model.

Initialize:

Choose a suitable integer $k \geq 1$ to obtain the desired success probability.

Process(a_i):

For $j = 1$ to k

Note: the next line makes two assumptions:

1. The random variable which is used to determine whether to set the set element should have its value determined separately for each iteration of the loop.
2. The value of i gives the counter of elements (i.e. for the b^{th} element being processed, we need that $i = b$).

With probability $\frac{1}{i}$, do $J_j \leftarrow a_i$

or $\frac{1}{i+1}$

The previous line could be implemented as follows:

$r \leftarrow Random(1, i)$

If $r = 1$ then

$J_j \leftarrow a_i$

End if

End for

Output:

Return the median of the set J.

Argument for correctness: we note that, in the original streaming algorithm for the median problem, the main property required is that, for each element added to the set J, the index of that element was picked from a uniform random distribution of size m *(i.e. each element from the stream was equally likely to be picked)*. The algorithm above replicates this property; to see why, we inductively argue the following claim:

Claim: after processing item a_j, $\Pr[J_\ell = a_i] = \frac{1}{j}$ for $1 \leq i \leq j$ for all $1 \leq \ell \leq k$.

Base case: if $j = 1$, then $\frac{1}{i} = 1$, and hence $\Pr[J_\ell = a_1] = 1$ for all ℓ.

Inductive step: if $j > 1$, then the probability of setting $J_\ell = a_j$ is equal to $\frac{1}{j}$. To see why $\Pr[J_\ell = a_i] = \frac{1}{j}$ for $1 \leq i < j$, we use the induction hypothesis; by the induction hypothesis, we have that $\Pr[J_\ell = a_i] = \frac{1}{j-1}$ immediately before processing element a_j. From the algorithm, it follows that, with probability $1 - \frac{1}{j} = \frac{j}{j} - \frac{1}{j} = \frac{j-1}{j}$, the item stored in J_ℓ is left unchanged. Hence, the probability of $\Pr[J_\ell = a_i] = \frac{1}{j-1}$ immediately after processing element a_j is given by $\frac{j-1}{j} \cdot \frac{1}{j-1} = \frac{1}{j}$. This proves the claim.

Hence, after processing each element, the probability of having taken a given element is given by a uniform random distribution. From this point on, it can be seen that the proof for lemma 9.6 (and hence, theorem 9.7) still holds for this new algorithm, given that we do not use an (asymptotically) increased amount of storage compared to the old algorithm.