CSE 6512 Randomization in Computing. Fall 2023
Exam I Solutions

---

1. Consider the following algorithm:

   repeat
   Pick a random $j \in [1, n]$;
   if $A[j] < 2$ then output "Type I" and quit;
   if $A[j] > 4$ then output: "Type II" and quit;
   forever;

   Analysis: Consider the case of A being of type I. The probability that $A[j] = 1$ on a radomly picked j is $\frac{1}{3}$. Thus the probability of quitting in any execution of the repeat loop is $\frac{1}{3}$. Therefore, the probability of failure in any execution of the repeat loop is $\frac{2}{3}$. As a result, the probability of failure in the first k iterations of the repeat loop is $\left(\frac{2}{3}\right)^k$. We want this probability to be no more than $n^{-\alpha}$. This happens when $k \geq \log_{3/2} n$. This implies that the run time of this algorithm is $\widetilde{O}(\log n)$, if the arracy is of type I. A similar analysis holds when the array is of type II.

2. Pick a random sample S of size $s = n^{2/3}$ from X. Identify and output the element q of S whose rank in S is $i\frac{s}{n}$. Sampling takes $O(s)$ time. q can be found from S in $O(s)$ time using the BFPRT selection algorithm.

   Let r_q be the rank of q in X. Using Sampling Lemma 2, *Prob.* $\left[|r_q - i| > \sqrt{3\alpha}\frac{n}{\sqrt{s}}\sqrt{\log n}\right] < n^{-\alpha}$. In other words, r_q is in the interval $i \pm O(n^{2/3}\sqrt{\log n})$ with a high probability, i.e, r_q is in the interval $i \pm O(n^{3/4})$ with a high probability.

3. Let $\mathcal{S}$ be a subset of the field. We pick a random element r of $\mathcal{S}$ and check if $F(r) = G(r)$. If not, we output "NO", else we output "YES". As was shown in class, the probability of an incorrect answer is $\leq \frac{n}{|\mathcal{S}|}$. This probability will be $\leq n^{-\alpha}$ if $|\mathcal{S}| \geq n^{\alpha+1}$.

   Also, $f_i(r)$ can be computed in $O(d_i)$ time, for $1 \leq i \leq k$. Thus $F(r)$ can be computed in $O(\sum_{i=1}^{k} d_i) = O(n)$ time. Similarly, $G(r)$ can also be computed in $O(n)$ time. Thus the total run time of the algorithm is $O(n)$.

4. Note that two matrices E and F are inverses of each other if $EF = FE = I$. Let $A = \Pi_{i=1}^{k} A_i$. In our problem, we have to check if $AC = CA = I$. We'll see how to check if $AC = I$. The same algorithm can be used to check if $CA = I$.

   Let S be a subset of the field with $|S| \geq n^{\alpha}$. Pick a random $n \times 1$ vector v each of whose elements is picked uniformly randomly from S. Compute ACv. If $ACv = v$, output 'yes' else output 'no'.

   Clearly, if $AC = I$, the algorithm will never output an incorrect answer. If $AC \neq I$ what is the probability that $ACv = v$? In other words, if $D = AC - I$, what is the probability that $Dv = 0$? Without loss of generality assume that the first row of D is non zero and the first

q entries of this row are nonzero and the rest of the elements are zero. Let d be this row. Let $d = (d_1\ d_2\ \ldots\ d_n)$ and $v^T = (v_1\ v_2\ \ldots\ v_n)$. $dv = 0$ if $v_1 = -\frac{\sum_{i=2}^{q} d_i v_i}{d_1}$. Now invoke the principle of deferred decisions and assume that all the entries of v have been chosen before v_1. Before v_1 is chosen, the value of $-\frac{\sum_{i=2}^{q} d_i v_i}{d_1}$ is fixed to be some value of S. (In fact $-\frac{\sum_{i=2}^{q} d_i v_i}{d_1}$ may not even be an element of S). Since v_1 is chosen uniformly randomly from S, the probability that v_1 equals $-\frac{\sum_{i=2}^{q} d_i v_i}{d_1}$ is no more than $\frac{1}{|S|} = n^{-\alpha}$.

Note that ACv can be computed with $(k+1)$ matrix-vector products. This will take $O(n^2 k)$ time.

5. Let h be the height of a random skip list $\mathcal{L}$ with n elements. It was shown in class that the height of $\mathcal{L}$ is $\widetilde{O}(\log n)$. Specifically, the height of $\mathcal{L}$ is $\leq c\alpha \log n$ with a probability of $\geq (1 - n^{-\alpha})$, for some constant c. The n elements in the data structure are at level 0. An element in level 0 goes to level 1 with probability $\frac{1}{2}$ and it does not go to level 1 with the same probability. An element in level 1 goes to level 2 with probability $\frac{1}{2}$ and it does not go to level 2 with the same probability, etc. This is how the skip list is constructed.

   Chernoff bounds imply that if μ is the mean of a binomial random variable X, then,

$$Prob.\left[X \geq \mu + \sqrt{3\alpha\mu \log_e n}\right] \leq n^{-\alpha}.$$

   Consider level k of $\mathcal{L}$ (where $1 \leq k \leq h$). The expected number of elements in this level is $n_k = \frac{n}{2^k}$. Using the Chernoff bounds, this number is $\leq N_k = \frac{n}{2^k} + \sqrt{3(\alpha+1)(n/2^k)\log_e n}$ with a probability of $\geq (1 - n^{-(\alpha+1)})$. $N_k \leq 2\frac{n}{2^k}$ with a probability of $\geq (1 - n^{-(\alpha+1)})$, for every level k, $1 \leq k \leq \frac{\log n}{2}$. The total number of nodes in the levels 1 through $\frac{\log n}{2}$ is thus $\leq \sum_{k=1}^{(1/2)\log n} \frac{n}{2^{k-1}} = O(n)$ with a probability of $\geq (1 - n^{-(\alpha+1)}(\log n)/2)$. Also, the number of elements in level k is $\leq 2\sqrt{n}$ with a probability of $\geq (1 - n^{-(\alpha+1)})$, for every $k \geq \frac{\log n}{2}$. This means that the total number of nodes in levels $\frac{\log n}{2} + 1$ through h is $O(\sqrt{n}\log n)$ with a probability of $\geq \left(1 - O(n^{-(\alpha+1)}\log n)\right)$.

   In summary, the total size of $\mathcal{L}$ is $O(n)$ with a probability of $\geq (1 - n^{-\alpha})$.

6. Note that when $m = n$, $h_{a,b}(x)$ simplifies to $(ax+b) \bmod p$. Fix x_1, x_2, y_1 and y_2. How many hash functions h are there in H under which $h(x_1) = y_1$ and $h(x_2) = y_2$? We observe that the following equations

$$(ax_1 + b) \bmod p = y_1$$

$$(ax_2 + b) \bmod p = y_2$$

   have a unique solution for a and b in $\mathcal{Z}_p$. There are a total of n^2 has functions in H. Thus it follows that $Prob.[h(x_1) = y_1 \text{ and } h(x_2) = y_2] = \frac{1}{n^2}$.