# Problem (1a)

Define $X_i$ as the value chosen in the $i$th iteration. It follows that

$$X = \frac{1}{k} \sum_{i=1}^{k} X_i$$

From this, we can compute $\mathbb{E}[X]$ and $\mathrm{Var}[X]$.

$$\begin{aligned}
\mathbb{E}[X] &= \frac{1}{k} \mathbb{E}\left[\sum_{i=1}^{k} X_i\right] \\
&= \frac{1}{k} \sum_{i=1}^{k} \mathbb{E}[X_i] \\
&= \frac{1}{k} \sum_{i=1}^{k} \mathbb{E}[X_i] \\
&= \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{n} z_j \frac{1}{n} \\
&= \frac{1}{k} \sum_{i=1}^{k} \sum_{j=1}^{n} z_j \frac{1}{n} \\
&= \frac{1}{k} \sum_{i=1}^{k} \alpha \\
&= \alpha
\end{aligned}$$

$$\mathrm{Var}[X] = \mathrm{Var}\left[\frac{1}{k} \sum_{i=1}^{k} X_i\right]$$

Observe that each $X_i$ is independent of another. Thus, the variance operator distributes:

$$\begin{aligned}
\mathrm{Var}[X] &= \frac{1}{k^2} \sum_{i=1}^{k} \mathrm{Var}[X_i] \\
\mathrm{Var}[X] &= \frac{1}{k^2} \sum_{i=1}^{k} \mathrm{Var}[X_i] \\
\mathrm{Var}[X_i] &= \mathbb{E}\left[X_i^2\right] - \mathbb{E}[X_i]^2 \\
\mathrm{Var}[X_i] &= \sum_{j=1}^{n} \frac{1}{n} z_j^2 - \alpha^2 \\
\mathrm{Var}[X_i] &= \sum_{j=1}^{n} \left(\frac{z_j^2 - \alpha^2}{n}\right) \\
\implies \mathrm{Var}[X] &= \frac{1}{k} \sum_{j=1}^{n} \left(\frac{z_j^2 - \alpha^2}{n}\right)
\end{aligned}$$

Since $\mathrm{Var}[X]$ is finite, we can apply Chebyshev's inequality:

$$\mathbb{P}\left[|X-\alpha| \geq \epsilon\right] \leq \frac{\sigma_x^2}{\epsilon^2}$$

$$\mathbb{P}\left[|X-\alpha| \geq \epsilon\right] \leq \frac{\frac{1}{k}\sum_{j=1}^{n}\left(\frac{z_j^2-\alpha^2}{n}\right)}{\epsilon^2}$$

$$\mathbb{P}\left[|X-\alpha| \geq \epsilon\right] \leq \frac{\sum_{j=1}^{n}\left(\frac{z_j^2-\alpha^2}{n}\right)}{k\epsilon^2}$$

$$\leq \frac{\delta\epsilon^2\sum_{j=1}^{n}\left(\frac{z_j^2-\alpha^2}{n}\right)}{(b-a)^2\epsilon^2}$$

$$\leq \frac{\delta\sum_{j=1}^{n}\left(\frac{z_j^2-\alpha^2}{n}\right)}{(b-a)^2}$$

To complete the proof, it suffices to show that:

$$\frac{\sum_{j=1}^{n}\left(z_j^2-\alpha^2\right)}{n(b-a)^2} \leq 1$$

Observe that the LHS is precisely

$$\frac{\mathrm{Var}\left[X_i\right]}{(b-a)^2}$$

$$= \frac{\sum_{i=1}^{n}\left(z_i-\alpha\right)^2}{(b-a)^2}$$

$$\leq 1$$

□

# Problem (1b)

A known generalization of the Chernoff inequality says that if $X_i \in [a_i, b_i]$, and if $X = \sum_{i=1}^{n} X_i$, then

$$\mathbb{P}\left[|X-\mu| \geq \Delta\right] \leq 2\exp\left\{-\frac{2\Delta^2}{\sum_{i=1}^{n}(b_i-a_i)^2}\right\}$$

Letting $\Delta = \epsilon, \mu = \alpha, a_i = a, b_i = b, n = k$, our bound is

$$\mathbb{P}\left[|X-\alpha| \geq \epsilon\right] \leq 2\exp\left\{-\frac{2\epsilon^2}{\sum_{i=1}^{k}(b-a)^2}\right\}$$

To find our desired value for $c$, we can bound the RHS to the right by $\delta$ and find an appropriate value for $c$

$$
\begin{aligned}
2\exp\left\{-\frac{2\epsilon^2}{\sum_{i=1}^{k}(b-a)^2}\right\} &\leq \delta \\
\exp\left\{-\frac{2\epsilon^2}{\sum_{i=1}^{k}(b-a)^2}\right\} &\leq \frac{\delta}{2} \\
-\frac{2\epsilon^2}{\sum_{i=1}^{k}(b-a)^2} &\leq \ln\left(\frac{\delta}{2}\right) \\
-\frac{2\epsilon^2}{k(b-a)^2} &\leq \ln\left(\frac{\delta}{2}\right)
\end{aligned}
$$

Since we care about the regime when $\delta < 1$, the right hand side can be assumed negative, and so we can perform the following inversion:

$$
\begin{aligned}
-\frac{k(b-a)^2}{2\epsilon^2} &\geq 1 \Big/ \ln\left(\frac{\delta}{2}\right) \\
k(b-a)^2 &\leq -2\epsilon^2 \Big/ \ln\left(\frac{\delta}{2}\right) \\
k(b-a)^2 &\leq 2\epsilon^2 \Big/ \ln\left(\frac{2}{\delta}\right) \\
k &\leq \frac{2\epsilon^2}{(b-a)^2 \ln\left(\frac{2}{\delta}\right)} \\
\frac{1}{k} &\geq \frac{(b-a)^2 \ln\left(\frac{2}{\delta}\right)}{2\epsilon^2}
\end{aligned}
$$

So let $c = 1/2$. $\langle$

**Problem 2.**

```
SORT(Array):
   bst ←MAKEBBST(Array)
   ⟨⟨ Store rank the number of elements to a given node's left ⟩⟩
   ⟨⟨ It takes n log n to make a BST  ⟩⟩
   ⟨⟨ and log n to determine the rank of an element chosen from the array ⟩⟩
   x ←Random number from 1 to n
   While bst[x].rank < n/4 or bst[x].rank > 3n/4:
         x ←Random number from 1 to n
   left, right ←PARTITION(Array)
   SORT(left)
   SORT(right)
```

The run time of this algorithm is the sum of the amount of time taken to choose a pivot at each stage, and the amount of comparisons made. Since a successful pivot produces (in reference to lecture 3), a lucky partition at every depth of the recursion, the amount of comparisons is $O(n \log n)$. We need to establish the cost required to choose pivots, however:

We first observe that there are, at the most, $\log_{4/3}(n)$ recursive depths (see lecture 3). Let us examine one level of the recursion, in order to make an expression for the total running time

$X$. At level $j$, there are, at the most, $2^j$ arrays being sorted and each has size at most $n(3/4)^j$. Each of these arrays has had a BST built for it. Moreover, each of these arrays keeps selecting a random number from its portion of the original array, until the obtained number has "lucky" rank. The latter process describes a geometric distribution with probability $1/2$ precisely. At the $j$th level, we index each of the geometric variables by $i$ from $i = 1$ to $i = 2^j$ with $X_{j,i}$. Putting all of this together, the cost at one level is

$$\log(n(3/4)^j) \sum_{i=1}^{2^j} (X_{ji}) + \left[n(3/4)^j \log\left(n(3/4)^j\right)\right](2^j)$$

Sum this over all levels but the last one to get:

$$\sum_{j=0}^{\log_{4/3}(n)-1} \log(n(3/4)^j) \sum_{i=1}^{2^j} (X_{ji}) + \left[n(3/4)^j \log\left(n(3/4)^j\right)\right](2^j)$$