CS 577: Introduction to Algorithms — Homework 7 Solutions

1

For simplicity, we let A be the list representing the evidence room, and x the positive integer (file) we want to find. Let T be a random variable that is equal to the number of accesses to files made by our search algorithm. Our task is to compute the expectation $\mathbf{E}[T]$. Let also X_i be a random variable that is equal to 1 if the i-th element of A is accessed and compared to *x at any round of the algorithm*. For example, if the length of the list is 8, after the execution of our algorithm, the array of the random variables X_i may look like

$$(0, 1, 0, 0, 0, 1, 0, 1)$$

Since we compare each item of the list with *x at most once*, the total number of accesses is simply

$$T = \sum_{i=1}^{n} X_i.$$

By linearity of expectation we obtain that $\mathbf{E}[T] = \sum_{i=1}^{n} \mathbf{E}[X_i]$. We have

$$\mathbf{E}[X_i] = 1\,\mathbf{P}[X_i = 1] + 0\,\mathbf{P}[X_i = 0] = \mathbf{P}[X_i = 1].$$

For a specific X_i at round k of our search algorithm, we have three possible outcomes

1. Item $A[i]$ is compared to x. In this case $X_i = 1$.
2. Item $A[i]$ is belongs to the part of A that is discarded. In this case $X_i = 0$.
3. Item $A[i]$ belongs to the part of A that we will continue our search. In this case the value of X_i will be determined in a later round.

We compute the probability that $X_i = 1$ conditioned on the event that X_i is determined in the k-th round of our algorithm. To simplify notation, let D_k denote the event that the value of X_i is determined at round k. Let j be the index of A that we are looking for, that is $A[j] \le x \le A[j+1]$. We have two cases

- $i \le j$.
  In this case, X_i is determined at round k if and only if Randy returns an element in the sublist $(A[i], \ldots, A[j])$. Furthermore, $X_i = 1$ if and only if Randy returns $A[i]$. Therefore, $\mathbf{P}[X_i = 1 | D_k] = \frac{1}{j-(i-1)} = \frac{1}{(j-i)+1}$.
- $i > j$.
  In this case, X_i is determined at round k if and only if Randy returns an element in $(A[j+1], \ldots, A[i]))$. Moreover, $X_1 = 1$ if and only if Randy returns $A[i]$. Therefore, $\mathbf{P}[X_i = 1 | D_k] = \frac{1}{i-j}$.

Notice that for all i, $\mathbf{P}[X_i = 1 | D_k]$ does not depend on k. Therefore,

$$\mathbf{P}[X_i = 1] = \begin{cases} \frac{1}{j-i+1}, & \text{if } i \le j \\ \frac{1}{i-j}, & \text{otherwise} \end{cases}$$

Now we have

$$\mathbf{E}[T] = \sum_{i=1}^{n} \mathbf{P}[X_i = 1] = \sum_{i=1}^{j} \frac{1}{j-i+1} + \sum_{i=j+1}^{n} \frac{1}{i-j} \le 2\sum_{i=1}^{n} \frac{1}{i} = 2H_n = O(\log n),$$

where by H_n we denote the n-th harmonic number.

2

a. Let X_i be the random variable that represents the length of the i^{th} arc. Using the hint, we observe that $X_1, \ldots, X_k$ are identically distributed, and therefore they have the same expectation

$$\mathbf{E}[X_1] = \mathbf{E}[X_2] = \ldots = \mathbf{E}[X_k]$$

Also by the linearity of expectation, we have

$$\sum_{i=1}^{k} \mathbf{E}[X_i] = \mathbf{E}\left[\sum_{i=1}^{k} X_i\right] = 1,$$

where the second equality is due to the fact that the lengths of k arcs would sum up to the circumference of the circle. This tells that $\mathbf{E}[X_i] = \frac{1}{k}$, for all $i \in \{1, \ldots, k\}$.

b. Once again we let x be the file we are looking for. There are two ways to access an element with $\mathrm{RANDY}()$ and by successive calls to $\mathrm{NEXT}()$ starting from some element that has already been obtained. Suppose we obtain k elements using $\mathrm{RANDY}()$ and sort them in increasing order to obtain the list $e_1, \ldots, e_k$. Let's see how we can use these elements to find x. Since the list is circular and sorted, we have that if x exists in the list, it must lie between two consecutive elements e_i, e_{i+1} (if $i = n$ we take the next element to be e_1). We can now use $\mathrm{NEXT}()$ to perform a linear search, and check all elements between e_i and e_{i+1}. Now we need to compute the expected number of elements between e_i and e_{i+1}. Intuitively, there are n/k elements on average between two elements accessed by $\mathrm{RANDY}()$. So we get that the total expected number of calls is k calls to $\mathrm{RANDY}()$ plus n/k calls to $\mathrm{NEXT}()$, $k + n/k$ overall. We can balance this out by choosing $k = \sqrt{n}$, to obtain $O(\sqrt{n})$ overall. To see why expected number of element between e_i and e_{i+1} is n/k we use question 2 (a). We know that if we choose k points on a circle with unit circumference uniformly at random, then the expected length of any arc is $\frac{1}{k}$. In this case, we have a circle with circumference n, and, by choosing k points, we have that the expected length of any single arc is $\frac{n}{k}$. Setting $k = \sqrt{n}$, we have that the expected length of any arc is $\frac{n}{k} = \frac{n}{\sqrt{n}} = \sqrt{n}$.

Algorithm 1: Randomized Search

Call $\mathrm{RANDY}()$ k times to obtain a list L of k elements.
Sort L in increasing order and get a list $e_1, e_2, \ldots e_k$.
Find e_i such that $e_i \leq x < e_{i \mod k+1}$
Starting at e_i follow the circularly linked list until $e_{i \mod k+1}$, by calling $\mathrm{NEXT}()$, returning true if x is found and false otherwise.