Average time of sequential search in ordered array

Example

Sequential search on an ordered array.

Find an item x in an ordered array I based only on comparisons of x to elements of I.

$I[0] \le I[1] \le I[2] \le \ldots \le I[k-1] \le I[k] \le \ldots \le I[n-1]$

Notation

size (I) - number of elements to be searched.

T (n) -
number of comparisons performed while searching of an entry in an n - element array I.

Average - case running time for successful sequential search (same as for an unordered array, which we have already done)

$$T_{avg}^{succ}(n) = \sum_{i=0}^{n-1} \Pr(I_i \mid \text{success}) * t(I_i) =$$

$$\sum_{i=0}^{n-1} \frac{1}{n}(i+1)$$

$$\frac{1+n}{2}$$

Note. For sequential search, $T_{avg}^{succ}(n)$ is the same for ordered and unordered array because sequential search does not take advantage of the order while successfully searching for a key.

For unsuccessful sequential search, there are n + 1 possible outcomes

x < I[0] - it results in 1 comparison

I[0] < x < I[1] - it results in 2 comparisons

...

I[i - 1] < x < I[i] - it results in i + 1 comparisons

...

I[n - 2] < x < I[n - 1] - it results in n comparisons

I[n - 1] < x - it results in n comparisons

Let's assume that all these outcomes are equally likely,

that is, all have the probability $\frac{1}{n+1}$.

$T_{avg}^{fail}(n) =$

$$\left(\sum_{i=0}^{n-1} \frac{1}{n+1}(i+1)\right) + \frac{1}{n+1} n$$

$$\frac{n}{2} + \frac{n}{1+n}$$

Hence, $T_{avg}(n) = q * T_{avg}^{succ}(n) + p * T_{avg}^{fail}(n) =$

$$= q * (1+n)/2 + (1-q) * \left(\frac{n}{2} + \frac{n}{1+n}\right) =$$

$$q * (1+n)/2 + (1-q) * \left(\frac{n}{2} + \frac{n}{1+n}\right)$$

$$\left(\frac{n}{2} + \frac{n}{1+n}\right)(1-q) + \frac{1}{2}(1+n)q$$

Simplify[%]

$$(3n + n^2 + q - nq)/(2 + 2n)$$

$$\text{Limit}\left[\frac{\frac{3n+n^2+q-nq}{2+2n}}{n}, n \to \infty\right]$$

$$\frac{1}{2}$$

$$0 < \frac{1}{2} < \infty$$

So, $T_{avg}(n) \in \Theta(n)$

More precisely,

$$T_{avg}(n) \sim \frac{1}{2} n$$

meaning

$$T_{avg}(n) = \frac{1}{2} n + o(n)$$

Optimality

For an ordered array, sequential search is NOT average - case optimal. For instance, binary search performs less comparisons on an average than sequential search does.

This part it optional
Begin of optional part

$$(3 n + n^2 + q - n q) / (2 + 2 n) - \frac{1}{2} n$$

$$-\frac{n}{2} + (3 n + n^2 + q - n q) / (2 + 2 n)$$

$$\text{Simplify}\left[-\frac{n}{2} + (3 n + n^2 + q - n q) / (2 + 2 n)\right]$$

$$\frac{2 n + q - n q}{2 + 2 n}$$

$$\text{Limit}\left[\frac{2 n + q - n q}{2 + 2 n}, \{n \to \infty\}\right]$$

$$\left\{1 - \frac{q}{2}\right\}$$

$$(3 n + n^2 + q - n q) / (2 + 2 n) - \frac{1}{2} n - \left(1 - \frac{q}{2}\right)$$

$$-1 - \frac{n}{2} + \frac{q}{2} + (3 n + n^2 + q - n q) / (2 + 2 n)$$

$$\text{Simplify}\left[-1 - \frac{n}{2} + \frac{q}{2} + (3 n + n^2 + q - n q) / (2 + 2 n)\right]$$

$$\frac{-1 + q}{1 + n}$$

Thus

$$T_{avg}(n) = \frac{1}{2} n + 1 - \frac{q}{2} + o(1)$$

End of optional part