CS32 H00 Handout—Page 1

In Section 12.1, p. 585 of DS4, we have the following formula:

$$\frac{1+2+3+4+5+6+7+8+9+10}{10} = 5.5$$

The author notes that this is the calculation of the average case running time of serial search of a 10 element array. The author explains how the formula is derived, and then goes on to show the derivation of a general formula of (n+1)/2 array accesses for the average case time of linear search of an n element array.

What the author does NOT say is that the formula makes two strong assumptions:

- That all elements in the array are equally likely
- That the element we are looking for will definitely be found.

The more general case for finding the average number of array accesses needed for an algorithm is an example of a technique you may learn in PSTAT120A called "finding the expected value of a random variable". It is one of the easier concepts from PSTAT120A to grasp.

Let X be the number of array accesses that is needed to find an element. We call X a random variable, because it can take on different actual values, x_1, x_2, x_3, etc. with various probabilities, p_1, p_2, p_3, etc.

The expected value of X, $E[X]$ is defined by this formula:

$$\mathrm{E}[X] = x_1 p_1 + x_2 p_2 + \cdots + x_k p_k \ .$$

The value x_1 is the outcome when we only have to look at `a[0]` and we find the element we are looking for. That's one array access, so x_1=1. The probability of that is 1/10, since we are assuming that finding what we are looking for in any given slot is equally likely (or to put it another way, every number in the array has an equal probability of being the one we are searching for.)
The next outcome, x_2 is when the array element is found in `a[1]`. If you think about it a moment, you'll see that the value of x_2=2 because we have to look at both `a[0]` and `a[1]`. In fact, x_3=3, and so forth. There are 10 outcomes, each with probability 1/10. So, the formula for expected value turns into:

$$E[X] = 1/10 + 2/10 + 3/10 + 4/10 + 5/10 + 6/10 + 7/10 + 8/10 + 9/10 + 10/10$$
$$= (1+ 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10)/10$$
$$= 55/10 = 5.5$$

But, what if it were not so simple? Suppose that half of the time, we don't find the number we are looking for at all. Then if each of the other 10 elements is still "equally likely", then each is found 5% of the time. (Note that the probabilities of all mutually exclusive outcomes for a single event need to sum to 100%).

On page 2 of this handout, we'll show how to find the expected number of array accesses (the "average case") for a more complicated situation than the one described above (and on p. 585) of your textbook. You'll need this for one of the homework problems.

CS32 H00 Handout—Page 2

"Average case" with different assumptions

Suppose we want to find the average number of array accesses for linear search, but this time we are NOT assuming that we always find the number.

In fact, suppose that half of the time, we don't find the number we are looking for at all. That is, with probability 0.5 (50%), we look through the entire array, and the number we are looking for isn't there.

The other half of the time we do find the number, and we still assume that each of the array elements is equally likely.

If each of the other 10 elements is still "equally likely", then each is found 5% of the time. (Note that the probabilities of all mutually exclusive outcomes for a single event need to sum to 100%).

That case of "not finding it at all" represents an *eleventh* possible outcome, x_{11}=10, with probability 50%, or 0.5. Now each of our other outcomes have probability 0.05 (5%) instead of 0.1. (10%)

Also note: x_{11} really is 10, not 11. That's not a typo. In lecture, or a future exam, I'm might ask you why, so think about why that is. Now, we have:

$$E[X] = (1(0.05)+2(0.05)+3(0.05)+4(0.05)+5(0.05)+6(0.05)+7(0.05)+8(0.05)+9(0.05)+10(0.05)) + (10)(0.5)$$
$$= (1+2+3+4+5+6+7+8+9+10)(0.05) + (10)(0.5)$$
$$= (55)(0.05) + (11)(0.5) = 2.75 + 5 = 7.75$$

With this example, you should now be able to calculate the answer to the following problem on the homework:

- Suppose there are nine array elements, `a[0]` through `a[8]`.
- Each of the odd elements, `a[1]`, `a[3]`, `a[5]`, and `a[7]` has a 10% chance (0.1) of being the one searched for.
- Each of the even elements `a[0]`, `a[2]`, `a[4]`, `a[6]` and `a[8]` has a 5% change (0.05) of being the one searched for.
- With probability 35% (0.35), the element being sought is not found in the array.

In this case, what is the expected number of array accesses?

Remember to set this up as an expected value problem. You need to determine how many possible outcomes there are, what the value of x is for each outcome, and multiply each value of x by the probability of that outcome.