EECS3101 notes :: A continuation on best/worst/average case complexity

Jerry Wu

Tues. January 24 2023

Example 1 :: Average case runtime

"motivation"-Larry YL Zhang 2023

A and B are the values of two dice rolled independently. We want to measure how many times the print line is executed. What is the average case runtime?

We want to find $E(X)$, where X is the number of times print is executed 10 times or once. What are the possible values of X?

$$X = 10, 1$$

Because the dice are uniform, and the events are independent, we can conclude that there are 36 possibilities. Out of those 36, there are exactly 6 that both dice rolls are equal. Therefore,

$$P(A = B) = \frac{1}{6}, P(A \neq B) = \frac{5}{6}$$

So,

$$E(X) = 10(\frac{1}{6}) + 1(\frac{5}{6}) = 2.5$$

Example 2 :: Worst case runtime (slide 9)

We have an outer loop of $O(n^2)$, and two inner loops of $O(n)$ and $O(\log(n))$. The worse of the inner loops is $O(n)$, so the worst case for this program would be $O(n^3)$.

> We can actually get rid of the exponent and base in log, so we can write any expression in terms of $alog_k(n^m)$ as $log(n)$ when working with big O. In other words, it's not based!

Example 3 :: Worst case runtime (slide 11)

"Never say never" - Larry YL Zhang 2023

In this question, we have an outer loop with $O(n)$, and an inner loop with $O(n)$, so we get $O(n^2)$, right? Well, this is the naive way of doing it. It's actually a tighter bound of $O(n)$ because for every key in the stack, there are 2 possible operations; push and pull. To add to this, the inner loop is dependent on the size of the stack.

Example 4 :: Average case runtime (slide 13)

- A_0 is picked from {0,1}
- A_1 is picked from {0,1,2}
- A_2 is picked from {0,1,2,3}
- A_n is picked from {0,1,2,. . . ,n+1}