13.6 Simplified Analysis Process

It's not that simple.

Summary of our (Painful) Analysis Process

- Construct a table of exact counts of all possible operations (takes lots of effort!)
- Convert table into worst case order of growth using 4 simplifications.

We will now propose an alternative method that avoids building a table altogether!

Our simplified analysis process will consist of:

- Choosing our cost model, which is the representative operation we want to count.
- Figuring out the order of growth for the count of our representative operation by either:
  - Making an exact count and discarding unnecessary pieces or...
  - Using intuition/inspection to determine orders of growth. This is something that comes with practice.

Example: Analysis of Nested For Loops - Exact Counts

Find the order of growth of the worst case runtime of `dup1`.

```
int N = A.length;
for (int i = 0; i < N; i += 1)
   for (int j = i + 1; j < N; j += 1)
      if (A[i] == A[j])
         return true;
return false;
```

We will choose our cost model to be the *number of == operations*.

Looking at the structure of the loops, the inner loop first gets run j=N-1 times. At the second iteration, i=1, so the inner loop runs an additional j=N-2 times. At the third iteration, i=2, so the inner loop runs an additional j=N-3 times. The total number of times the loop is run is thus:

$$\text{cost} = 1 + 2 + 3 + \ldots + (N-2) + (N-1)$$

This cost can be simplified to $\frac{N(N-1)}{2}$ (how?). We can use simplification to throw away all lower order terms and constants to get the worst case order of growth N^2.

Example: Analysis of Nested For Loops - Geometric Argument

- We can see that the number of equals can be given by the area of a right triangle, which has a side length of $N-1$.
- Therefore, the order of growth of area is N^2.
- This is definitely not something that is immediately obvious. It takes time and practice to see these patterns!

Previous
13.4 Asymptotic Behavior

Next
13.7 Big-Theta

Last updated 1 year ago