MA3105: Numerical Analysis

Satvik Saha, 19MS154 August 12, 2021

Consider the following code block, and suppose that the commented c_i's represent the time cost of executing the corresponding statement.

```
bool unique (int a[]) {
    for (i = 0; i <= n - 2; i++)          // c_1
        for (j = i + 1; i <= n - 1; j++)  // c_2
            if (a[i] == a[j])             // c_3
                return false;
    return true;
}
```

It is clear that the first `for` statement is executed n times; the interior is executed $n-1$ times. Similarly, the second `for` statement is executed $n-i$ times, with the interior executed $n-i-1$ times. The `if` block is executed the full $n-i-1$ times every loop at worst, 1 time at best. Thus, the total cost can be estimated as

$$\begin{aligned}
T(n) &= \sum_{i=0}^{n-1} c_1 + \sum_{i=0}^{n-2}\sum_{j=i+1}^{n} c_2 + \sum_{i=0}^{n-2}\sum_{j=i+1}^{n-1} c_3 \\
&= c_1 n + \sum_{i=0}^{n-2} c_2(n-i) + \sum_{i=0}^{n-2} c_3(n-i-1) \\
&= c_1 n + (c_2+c_3)\sum_{i=0}^{n-2}(n-i) - \sum_{i=0}^{n-2} c_3 \\
&= c_1 n + (c_2+c_3)n(n-1) - \frac{1}{2}(c_2+c_3)(n-1)(n-2) - c_3(n-1).
\end{aligned}$$

By expanding further and collecting coefficients, we have

$$\begin{aligned}
T(n) &= c_1 n + (c_2+c_3)n^2 - (c_2+c_3)n - \frac{1}{2}(c_2+c_3)n^2 + \frac{3}{2}(c_2+c_3)n - (c_2+c_3) - c_3 n + c_3 \\
&= \frac{1}{2}(c_2+c_3)n^2 + \frac{1}{2}(2c_1+c_2-c_3)n - c_2.
\end{aligned}$$

As $n \to \infty$, we see that $T(n)$ grows quadratically. We write $T(n) \in O(n^2)$.

Lemma. *Given any polynomial*

$$f(n) = a_k n^k + a_{k-1}n^{k-1} + \cdots + a_1 n + a_0$$

with at least one non-zero coefficient a_i, we can write $f(n) \in O(n^k)$.

Proof. Set $M = |a_k| + |a_{k-1}| + \cdots + |a_0|$. Then for all $n \geq 1$, the triangle inequality gives

$$\begin{aligned}
|f(n)| &= \left|a_k n^k + a_{k-1}n^{k-1} + \cdots + a_1 n + a_0\right| \\
&\leq |a_k|n^k + |a_{k-1}|n^{k-1} + \cdots + |a_1|n + |a_0| \\
&\leq |a_k|n^k + |a_{k-1}|n^k + \cdots + |a_1|n^k + |a_0|n^k \\
&= Mn^k.
\end{aligned}$$

□