CIS 575. Introduction to Algorithm Analysis
Material for January 26, 2024

Sum Estimates for Loop Analysis

1 Analyze Loops by Estimating Sums

We shall prepare for a general result for analyzing the running time of (nested) iterative constructs, in particular **for** loops but often applicable even for **while** loops.

First observe (with n the input size) that if for a given loop, there exists functions f and g such that

- the loop iterates $g(n)$ times
- the i'th iteration runs in time $f(i)$

then the total running time is given by the sum

$$\sum_{i=1}^{g(n)} f(i)$$

For example, consider the nested loop

$$\begin{array}{l} \textbf{for } j \leftarrow 1 \textbf{ to } n^2 \\ \quad \textbf{for } k \leftarrow 1 \textbf{ to } j^3 \\ \quad\quad q \leftarrow q + j + k \end{array}$$

The inner loop iterates j^3 times, with each iteration taking constant time; hence the running time of the inner loop is given by (or rather is proportional to) the sum

$$\sum_{i=1}^{j^3} 1 = \mathbf{j^3}$$

The outer loop iterates n^2 times. In the i'th iteration, j will equal i and hence the running time of that iteration is (proportional to) i^3. Thus the total running time is given by the sum

$$T_0(n) = \sum_{\mathbf{i=1}}^{\mathbf{n^2}} \mathbf{i^3}$$

We would like to know if this is in $\Theta(n^p)$ for some p.

It is easy to see that n^8 is an upper bound, as we can approximate i by its *highest* possible value, n^2:

$$T_0(n) = \sum_{i=1}^{n^2} i^3 \leq \sum_{i=1}^{n^2} (n^2)^3 = \sum_{i=1}^{n^2} n^6 = n^2 \cdot n^6 = n^8$$

What about a lower bound? If we approximate i by its *lowest* possible value, 1, we get

$$T_0(n) = \sum_{i=1}^{n^2} i^3 \geq \sum_{i=1}^{n^2} 1 = n^2$$

Thus $T_0(n)$ is sandwiched between n^2 and n^8, not exactly a narrow estimate; we would surely want a much more precise analysis. Skilled mathematicians are able to find the exact value[1] of $T_0(n)$, but for us it suffices to express it using big Theta notation.

And indeed, it turns out that $\mathbf{T_0(n) \in \Theta(n^8)}$ since we can prove that $T_0(n) \in \Omega(n^8)$; of course we cannot prove $T_0(n) \geq n^8$ but we can prove

$$T_0(n) \geq \frac{n^8}{16}$$

as demonstrated by the calculation (where $\lceil x \rceil$ denotes the *ceiling* of x, such as $\lceil 4.62 \rceil = 5$):

$$T_0(n) = \sum_{i=1}^{n^2} i^3 \geq \sum_{i=\lceil \frac{n^2}{2} \rceil}^{n^2} i^3 \geq \sum_{i=\lceil \frac{n^2}{2} \rceil}^{n^2} (\frac{n^2}{2})^3 = \sum_{i=\lceil \frac{n^2}{2} \rceil}^{n^2} \frac{n^6}{8} \geq \frac{n^2}{2} \cdot \frac{n^6}{8} = \frac{n^8}{16}$$

This is a general result, under certain assumptions, as we shall show in the next notes.

1

$$\text{Since } \sum_{i=1}^{m} i^3 = \frac{m^2(m+1)^2}{4} \text{ we see that } T_0(n) = \frac{n^8 + 2n^6 + n^4}{4}$$