• Calculating the Efficiency of Binary Search
• Describing the Efficiency of Computations

Overview

March 13, 2000

Comp 212
characteristics of the input, such as length. Examining the program text, however, cost depends on

For a few programs, the cost is fixed and can be calculated by

$\text{time steps, rather than } 2^n \text{ versus } 3^n \text{ steps.}$

to gross differences in time or space consumed, for example, $n^2$ versus $n$. When comparing programs (or algorithms), you should first pay attention

instead of seconds and bits.

A running program consumes resources such as time (seconds) and space

Efficiency of Computations
term.

\[ u^2 \text{ i.e., they don't grow faster than } k n^2, \text{ where } k \text{ is some constant} \]

we mean that \( p \) equals \( n^3 \), plus some terms that are \( \ll n^3 \) on the order of

\[ (n^2)O + n = O \]

is less than (some multiple of the function \( (u)f \)). For example, if we say

\( \text{The Big-Oh notation simply means that the cost function is bounded by} \)

\( \text{function over the positive integers.} \)

\( \text{Big-Oh, and is always of the form } \text{order-of-magnitude} \text{ of the cost. The notation used is sometimes called} \)

\( \text{When we make gross comparisons of programs, we often refer to the} \)

Order Arithmetic
for all \( u < u_0 \).

(4) \((u)f \geq (u)\eta \geq 0\)

exist positive integers \( c \) and \( u_0 \) such that

In other words, \((u)f\) \(\in\) the set of all functions \((u)\eta\) such that there

for all \( u < u_0 \).

(3) \((u)f \geq (u)\beta \geq 0\)

if there is a positive integer \( c \) and \( u_0 \) such that

(2) \(((u)f)O = (u)\beta\)

Definition. A function \((u)f\) \(\in\) is said \(\alpha\) to be \((u)\beta\) written

More precisely,

Order Arithmetic (cont.)
(7) \[(\z u)O = u + \cdots + 3 + z + I\]

(6) \[(u)O + \frac{z}{z u} = u + \cdots + 3 + z + I\]

(5) \[\frac{z}{u} + \frac{z}{z u} = \frac{z}{(1 + u)u} = u + \cdots + 3 + z + I\]

For example,

Order Arithmetic (cont.)
(II) \[ ((u)b \times (u)f)O = ((u)b)O \times ((u)f)O \]

(6) \[ ((u)f)O = ((u)f)O + ((u)f)O \]

(6) \[ ((u)f)O = ((u)f)O \times \lambda \]

(8) \[ ((u)f)O = (u)f \]

Involving operators of magnitude quantities:

Here are some equivalences that allow you to manipulate equations:

Order Arithmetic (cont.)
changes the value by a constant factor of log₂c.

Because changing the base of the logarithm doesn’t affect the order of magnitude, the base to which a logarithm is computed doesn’t affect the order of arithmetic (cont.)
The longest traversal is \( \log(n + 1) \) where \( n \) is the length of the array.

Consider the following array and its possible traversals by findIndex.

Calculating the Efficiency of Binary Search
- Specifically, if I asked you find "Alan Cox" in the phone book would you start in the middle?

- How do you find a number in a phone book?

Can we do better than binary search?
\[
\text{mid} = 10 + \frac{10 - \text{key} \times (a[11] - a[10])}{(a[10] - a[9])} \\
\text{mid} = 10 + \frac{10 - \text{key} \times (a[11] - a[10])}{(a[10] - a[9])} \\
\]

What we're looking for and replace \((a[11] - a[10])\) with an expression that places us closer to

\[
\text{mid} = 10 + \frac{10}{2} \\
\text{as}
\]

\[
\text{mid} = 10 + \frac{10}{2} \\
\]

We can rewrite

**Interpolation Search**