CS 240 Module 1: Introduction and Asymptotic Analysis

Graham Cooper

May 5th, 2015

What is this course about?

<u>Data Structures</u>

- hand in hand with algorithms
- patterns for storing/maintaining large data
- perform operations on the data (insertion, deletion, sorting)
- algorithms for these operations

<u>Example, Student Records</u>

- Abstract Data Type → a concept (eg. Stack)
- Data Structure → implementation of a data type (eg, Linked List)

Algorithms are presented using pseudocode and analyzed using order notation

Dictionary ADT

- a set of items with the following operations
- insertion, deletion, search

Problems

Problem of Sorting

Given a $\overset{input}{set/array}$ of $\overset{size}{sizen}$ of numbers, $\overset{operation(output)}{arrangetheminincreasingorder}$

Solution for a problem - Algorithm

Different solutions/algorithms eg. bubble sort/quicksort to solve sorting problems
The Algorithm should be correct and efficient(Time complexity and space).
RAM Random Access Memory: No matter where you are accessing memory, it takes the same about of time
This can also be thoguht of as a random access machine which is an abstract machine

$$f(n) \in O(g(n)) \overset{informal}{=} f(n) \leq g(n)$$

$$f(n) \in \Omega(g(n) \overset{informal}{=} f(n) \geq g(n)$$

$$f(n) \in \Theta(g(n)) \overset{informal}{=} f(n) = g(n)$$

$$f(n) \in o(g(n)) \overset{informal}{=} f(n) < g(n)$$

$$f(n) \in \omega(g(n)) \overset{informal}{=} f(n) > g(n)$$

Number of primitives plus r.a.o. time complexity

$$f(n) = 100n + 200$$

$$g(n) = 1/2n^2$$

$$Show f(n) \in O(g(n))$$

$$\exists c, 100n + 200 \leq c \cdot 1/2n^2$$

$$n \geq n_0$$

$$Lets have c = 1/2$$

$$100n + 200 \leq 1/2n^2 \cdot 1.2$$

$$100/n + 200/n^2 \leq 1/2 \cdot 1/2$$

$$if n > 200$$

$$100/n + 200/n^2 < 100/200 + 200/200^2 \leq |1/2 + 1/2| = 1 \cdot 1/2$$

Order Notation

- May 7th
$f(n) \leftarrow O(g(n)) \iff \exists c, n_0\ \forall n > n_0\ 0 < f(n) < cg(n)$
$f(n) = 2n^2 + 3n + 11\ g(n) = n^2$
prove $f(n) \in O(g(n))$
We need to find c,n_0 such that $\forall n > n_0$

$$2n^2 + 3n + 11 < c \cdot n^2$$

$$\iff 2 + \frac{3}{n} + \frac{11}{n^2} < c$$

$$n_0 = 1 \implies c > 2 + 3 + 11 = 16$$

$$n_0 = 2 \implies c > 2 + 3/2 + 11/4$$

$\therefore n_0 = 1$ and $c = 16$

$$f(n) = 2010n + 1388n$$

$$g(n) = n^3$$

prove $f(n) \in o(g(n))$
find $n_0 > 0$, such that for all $c > 0$ $2010n^2 + 1388n < cn^3$
I can express n as a function of c

$$cn^3 - 2010n^2 - 1388n > 0$$

$$cn^2 - 2010n - 1388 > 0$$

$$\delta = 2010^2 + 4 * c * 1388$$

$$n > n_0 = \frac{2010 + \sqrt{2010^2 + 4 * c * 1388}}{4}$$

[IMAGE 2]
show $f(n) \in \Theta(g(n))$
$f(n) = n + 2\sqrt{n}log(n)\ g(n) = n$
$log(n) \in o(\sqrt{n})$
$\sqrt{n}log(n) \in o(\sqrt{n} \times \sqrt{n}) = o(n)$
We need to find c_1 c_2, n_0 such that

$$c_1 \times n \leq n + 2\sqrt{n}logn(n) \leq c_2 \times n$$

$\forall n > n_0$
$c_1 = 1$
for finding c_2

$$n + 2\sqrt{n}log(n) \leq c_2 \times n$$
$$1 + \frac{2\sqrt{n} \times log(n)}{n} \leq c_2$$
$$n_0 = 64$$
$$c_2 \geq 1 + \frac{2\sqrt{64}log(64)}{64} = 1 + \frac{2 \times 8 \times 6}{64} > 3$$
$$c_2 = 3works$$

$c_1 = 1$ $c_2 = 3$ $n_0 = 64$
Could use any value not just 64

Algebraic	Asymptotic
$f(n) = g(n)$	$f(n) \in O(g(n))$
$f(n) < g(n)$	$f(n) \in o(g(n))$
$f(n) > g(n)$	$f(n) \in \omega(g(n))$
$f(n) \leq g(n)$	$f(n) \in O(g(n))$
$f(n) \geq g(n)$	$f(n) \in Theta(g(n))$

$$f(n) \in \theta(1)$$
$$f(n) \in \theta(log(n))$$
$$f(n) \in \theta(log^k n)$$
$$f(n) \in \theta(\sqrt{n})$$
$$f(n) \in \theta(n)$$

example, Slide 33
$f(n) = log(n)$ $g(n) = n^i$

$$\lim_{n/to/infty} \frac{f(n)}{g(n)} = \lim_{n/to/infty} \frac{log(n)}{n^i}$$
$$(derivitive) = \frac{c \cdot \frac{1}{n}}{i \times n^{i-1}} = \frac{c}{i \times n^i} \implies 0$$
$$\implies f(n) \in o(g(n))$$

Order Notation

- May 12th
(slide 33)

$$\lim_{n\to\infty} \frac{logn}{n^i}$$

$$\lim_{n\to\infty} \frac{\frac{1}{n}}{in^{i-1}}$$

$$log(n) \in o(n^i)$$

(Slide 34)

$$\forall n > n_0$$

$$n < \underbrace{(2 + sin(n7/2))}_{[1,3]} \leq 3n$$

$$c_1 g(n) \leq f(n) \leq c_2 g(n)$$

$$g(n) = n$$

(Slide 36)

$$f(n) \in O(g(n))$$

$$\implies \exists c, n_0, \forall n > n_0, f(n) \leq c_1 g(n)$$

$$h(n) \in O(f(n) + g(n))$$

$$\exists c, n_0, \forall n > n_0$$

$$h(n) \leq c(f(n) + g(n)) \leq c(2 \cdot max\{f(n), g(n)\} \leq 2c(max \cdot \{f(n), g(n)\})$$

$$f(n) = n^2 + \sqrt{n} log^{1000} n + n$$

$$f(n) \in O(max\{n^2, \sqrt{n} log^{1000} n + n\})$$

$$f(n) \in O(n^2)$$

$$[formallyprove \sqrt{n} log^{1000} n \in o(n^2)]$$

$$\forall n > n_0 f(n) < c_1 g(n)$$

$$\exists c_1, c_2, n_0 g(n) < c_2 h(n) \to f(n) < c_1 c_2 h(n)$$

(slide 37)

Arithmetic Sequence

$$\sum_{i=0}^{n-1}(a+di) = \sum_{i=0}^{n-1} a + \sum_{i=0}^{n-1}(di)$$

$$= an + d\sum_{i=0}^{n-1} i$$

$$= an + d\frac{n(n-1)}{2} \in \Theta(n^2)$$

$$0+1+2+...+(n-1) = \frac{n(n-1)}{2}$$

k=2

$$1+4+9...+n^2 = \frac{n(n+1)(2n+1)}{6} \in \Theta(n^3)$$

k=3

$$1+8+27...+n^3 = (\frac{n(n+1)}{2})^2 \in \Theta(n^4)$$

k=4

$$1+16+82...+n^4 = \Theta(n^5)$$

k=x

$$\sum_{i=1}^{n}(a+di^k) \in \Theta(n^{k+1})$$

Geometric Sequence

$$\sum_{i=1}^{n} ar^i \in \Theta(r^n) for r > 1$$

$$\in \Theta(n) for r = 1$$

$$\in \Theta(1) for r < 1$$

$$r = 2 : a(1 + 2 + 4 + 8... + 2^n) \in \Theta(2^n)$$
$$r = 1/2 : a(1 + 1/2 + 1/4 + 1/8...) < 2a \in \Theta(1)$$

Harmonic Sequence

$$\sum_{i=1} n\frac{1}{i} \in \Theta(logn)$$
$$ln(n+1) < \sum_{i=1}^{n} log(i) < ln(n) + 1$$
$$ln(n) = \frac{log_2 n}{log_c n} \in \Theta(logn)$$

Misc Math Facts
(slide 38)

$$\sum_{i=1}^{n} ir^i \in \Theta(nr^n)$$
$$1 + 1/2 + 1/4 + 1/8... + 1/n <\approx 2$$
$$1 + 1/4 + 1/9 + 1/16 + ... + 1/n^2 <\approx \frac{\pi^2}{6}$$

$$n! \in o(n^n)$$
$$logn! \in log(n^n) = \Theta(nlogn)$$

Loop Analysis

slide 41

$$T(n) = c_1 + \sum_{i=1}^{n}\sum_{j=i}^{n} c_2$$
$$= c_1 + \sum_{i=1}^{n}(n - i + 1)c_2$$

$$= c_1 + c_2 \sum_{k=1}^{n} k$$

$$= c_1 + c_2 \times \frac{n(n+1)}{2} \in \Theta(n^2)$$

SLide 42

$$T(n) = c_1 + \sum_{i=1}^{n} \sum_{j=i}^{n} (c_2 + \sum_{k=i}^{j} c_3)$$

$\sum_{k=i}^{j} c_3 = (j - i + 1) \times c$ aside*

$$= c_1 + \sum_{i=1} n \underbrace{\sum_{j=i}^{n} (c_2 + (j - i + 1)c_3)}_{A}$$

$A = \sum_{j=i}^{n} c_2 + c_3 \sum_{j=i}^{n} (j - i + 1)$ Aside* let $t = j - i + 1$
$A = (n - i + 1)c_2 + c_3 \sum_{t=1}^{n-i+1} t$
$A = (n - i + 1)c_2 + c_3 \frac{(n-i+1)(n-i+2)}{2}$

$$= c_1 + \sum_{i=1}^{n} ((n - i + 1)c_2 + c_3 + \frac{(n - i + 1)(n - i + 2)}{2})$$

Let L = n-i+1

$$= c_1 + \sum_{L=1}^{n} (Lc_2 + c_3 \frac{(L)(L+1)}{2})$$

$$= c_1 + \sum_{L=1}^{n} (\frac{c_3}{2} L^2 + (c_2 + \frac{c_4}{2})L)$$

$$= c_1 \frac{c_3}{2} \times \frac{n(n+1)(2n+1)}{6} + (c_2 + \frac{c_3}{2} \frac{n(n+1)}{2}$$

$$= \frac{1}{6} \times \frac{c_3}{2} n^3 + o(n^3) \in \Theta(n^3)$$

Slide 43

$$T(A_4) = c_1 + \sum_{i=1}^{n} (c_2 + \log i)$$

$$= c_1 + c_2 \times n + \sum_{i=1} n \log i$$

Aside: $\log n! \in \Theta(nlogn)$

$$= \Theta(n \log n) + o(n \log n) \in \Theta(n \log n)$$

MergeSort

5 2 1 4 |6 8 0 3
1 2 4 5 |0 3 6 8
0 1 2 3 4 5 6 8

Analysis of Mergesort

T(n) = $\Theta(1)$ if n = 1
$T(ceil(\frac{n}{2})) + T(floor(\frac{n}{2})) + \Theta(n)$ for n > 1

If n is a power of 2

$$T(ceil(\frac{n}{2})) + T(floor(\frac{n}{2})) + \Theta(n)$$

$$= 2T(\frac{n}{2}) + cn$$

$$= 4T(\frac{n}{4}) + 2 \times c\frac{n}{2} + cn$$

$$= 8T(\frac{n}{8}) + 4 \times c\frac{n}{4} + 2 \times c\frac{n}{2} + cn$$

After logn strps

$$= 2^{logn} \times T(1) + cn + cn + ... + cn$$

$$= n \times d_c n + \log n$$

$$\in \Theta(n \log n)$$