CS223—Data Structures and Algorithms

Lecture 1
Jan 23, 2001

Textbook: Introduction to algorithms
by Cormen, Leiserson, Rivest and Stein

Textbook: Algorithms in JAVA (part 5 on Graph Algorithms)
by Sedgewick

0. About CS223

- Course homepage: http://www.cs.montana.edu/bhz or http://www.cs.montana.edu/courses/223.
- We will cover elementary data structures, discrete mathematics and algorithm design.
- If you want to know the overall contents of the course, check the old 223 homepage at http://www.cs.montana.edu/courses/spring2006/223. But the details of the two courses might be slightly different.
- Evaluation: in-class tests (30%), lab assignments (30%) and final exam (40%)
- Evaluation: in-class tests (30%), lab assignments (40%) and final exam (30%)
- To pass the course, you must get at least 30 out of 100 in the final exam.

1. Overview

- Basic concepts on data structures and algorithms
- Solving recurrence relations
- Elementary probability
- Balanced binary search trees
- Searching and hashing techniques
- Graph algorithms and applications
- Some more advanced topics to be determined

2. Review of algorithmic complexity

- The measure of efficiency of a program (algorithm) $f(n)$ is called *algorithmic complexity* of $f(n)$.

- O-notation. $O(g(n))$ is defined as the set of all functions $f(n)$ such that there exist positive constants c, N and $f(n) \leq c \cdot g(n)$ for all $n \geq N$.

- Properties of O-notations

- Ω-notation. $\Omega(g(n))$ is the set of all functions $f(n)$ such that there exist positive constants c, N and $f(n) \geq c \cdot g(n)$ for all $n \geq N$.

- Θ-notation. $\Theta(g(n))$ is the set of all functions $f(n)$ such that there exist positive constants c_1, c_2, N and $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ for all $n \geq N$.

3. Establishing Order Relationships

- Notice that the notion of order is restricted to real-valued functions $f(n) : N \rightarrow R$ that are eventually positive; i.e., there exists an integer n_0 such that $f(n) > 0$ for all $n > n_0$. Let $\mathcal{F}$ be the set of such functions.

- Given $f(n), g(n) \in \mathcal{F}$, we say that $f(n)$ has *smaller order than* $g(n)$ if $O(f(n)) \subset O(g(n))$, i.e., $O(f(n))$ is strictly contained in $O(g(n))$.

- Let $P(n) = a_k n^k + a_{k-1} n^{k-1} + ... + a_1 n + a_0$ be any polynomial of degree k, then $P(n) \in \Theta(n^k)$.

- Let $f(n), g(n) \in \mathcal{F}$. Let the limit of $\frac{f(n)}{g(n)}$ be L as $n \to \infty$, i.e., $L = \lim_{n\to\infty}\frac{f(n)}{g(n)}$, then the following results hold.
  - If $0 < L < \infty$, then $f(n) \in \Theta(g(n))$.
  - If $L = 0$, then $O(f(n)) \subset O(g(n))$.
  - If $L = \infty$, then $O(g(n)) \subset O(f(n))$.

- Let $f(n), g(n) \in \mathcal{F}$. We define $o(g(n))$ as the set of all functions $f(n)$ such that $\lim_{n\to\infty}\frac{f(n)}{g(n)} = 0$.

- Let $f(n), g(n) \in \mathcal{F}$. We say $f(n)$ is strongly asympotic to $g(n)$ if $\lim_{n\to\infty}\frac{f(n)}{g(n)} = 1$. (Certainly $f(n) \in \Theta(g(n))$.)

- L'Hôpital's Rule: Let $f(x)$ and $g(x)$ be functions that are differentiable for sufficiently large real numbers x. If $\lim_{x\to\infty} f(x) = \infty$ and $\lim_{x\to\infty} g(x) = \infty$, then
$\lim_{x\to\infty}\frac{f(x)}{g(x)} = \lim_{x\to\infty}\frac{f'(x)}{g'(x)}$.