CSE 373 Spring 2006

Data Structures and Algorithms
Useful Mathematical Facts

Notation

$\lfloor x \rfloor$ floor function: the largest integer $\leq x$
$\lceil x \rceil$ ceiling function: the smallest integer $\geq x$

Positional Number System

In base 10, an unsigned integer is $x = \sum_{i=0}^{i=n} a_i 10^i$, where a_i is a digit from 0 to 9.
In base 2 (binary) an unsigned integer is $x = \sum_{i=0}^{i=n} a_i 2^i$, where a_i is a bit value, 0 or 1.

Logs

$log_2 x = y$ means $x = 2^y$
$log(x.y) = logx.logy$; $\quad log(x/y) = logx - logy$; $\quad log(x^y) = y.logx$
$loglogx < logx < x$ for all $x > 0$
$log_x a = \frac{log_2 a}{log_2 x}$

Series

S(n) = 1 + 2 + 3 + ... + n = $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$
C(n) = $1 + 2^2 + 3^2 + ... + n^2 = \sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$
H(n), the n^{th} harmonic number $= 1 + 1/2 + 1/3 + ... + 1/n = ln\ n + c$ where $ln\ n$ is the natural logarithm of n and c is a constant. Thus H(n) = O(logn)

The Big-Oh Notation

$T(n) = O(f(n))$ if there are constants c and n_0 such that $T(n) \leq c\ .\ f(n)$ for all $n \geq n_0$

Constant time $O(1)$
Logarithmic time $O(logn)$
Linear time $O(n)$
$O(nlogn)$ grows faster than linear time but not as fast as quadratic time
Quadratic time $O(n^2)$
Cubic time is $O(n^3)$
Polynomial time is $O(n^k)$ for some k
Exponential time is $O(c^n)$ for some $c > 1$