

Caijie Zhang

<http://www.cs.ucsb.edu/~caijie>**1. General Rules for Running Time Calculations(2.4.2)****(1) RULE1-FOR LOOPS:**

*The running time of a for loop is at most the running time of the statements inside the for loop (including tests) times the number of iterations*

**(2) RULE2-NESTED LOOPS:**

*Analyze these inside out. The total running time of a statement inside a group of nested loops is the running time of the statement multiplied by the product of the sizes of all the loops*

**(3) RULE3-CONSECUTIVE STATEMENTS:**

*These just add (which means that the maximum is the one that counts; see rule 1 on page 42).*

**(4) RULE4-IF/ELSE**

if ( condition )

    S1

else

    S2

*The running time of an if/else statement is never more than the running time of the test plus the larger of the running times of S1 and S2.*

**(5) RECURSIVE FUNCTIONS**

- I. *If the recursion is really just a thinly veiled for loop, the analysis is usually trivial.*
- II. *When recursion is properly used, the analysis will involve a recurrence relation that needs to be solved.*

**2. Hashing (5)**

- a) **Separate chaining:** *Separate chaining keeps a list of all elements that hash to the same value.*
- b) **Linear Probing:** *Linear Probing is an open addressing method to solve collisions in a hash table. If a collision occurs, it linear probes the alternative cells in the hash table until an empty cell is found.*

Example:

Keys : {89,18,49,58,69,21,50}

$H(x) = x \bmod 10$