COT 6405
ANLYSIS OF ALGORITHMS

A Survey of Common Running Times

Computer & Electrical Engineering and Computer Science Department
Florida Atlantic University

Common order of growth functions

Asymptotic Notation	Description
$\Theta(1)$	constant
$\Theta(\lg \lg n)$	log log
$\Theta(\lg n)$	log
$\Theta(n^c)$, $0 < c < 1$	sublinear
$\Theta(n)$	linear
$\Theta(n\lg n)$	nlogn
$\Theta(n^2)$	quadratic
$\Theta(n^3)$	cubic
$\Theta(n^k)$, $k \geq 1$	polynomial
$\Theta(c^n)$, $c > 1$	exponential
$\Theta(n!)$	factorial
$\Theta(n^n)$	

Sublinear Time

- RT is asymptotically smaller than linear
- Reading the input takes linear time
- Occurs when:
  - Input “queried” indirectly rather than read completely
  - Try to minimize the number of queries

- <u>Example problem</u>

Given a sorted array A of n numbers, determine whether a given number x belongs to the array.

- traverse the array $\Rightarrow \Theta(n)$
- the *binary search algorithm* takes $\Theta(\log n)$

Linear Time $\Theta(n)$

- One-pass or a constant number of passes through the input elements
- Example problems
  - Find the max/min of *n* numbers
  - Merge two sorted arrays into one sorted array

O(nlogn) Time

- A very common RT
- Any algorithm that splits its input into two equal-sized pieces, solve each piece recursively, then combine the two solutions in linear time
- Example: Merge-Sort algorithm

$$T(n) = 2 \cdot T(n/2) + \Theta(n)$$

$$T(n) = \Theta(n\log n)$$

Quadratic Time, $\Theta(n^2)$

<u>Example problem</u>

Given n points in the plane, find the closest pair of points.

for each input point (x_i, y_i)
  for each other input point (x_j, y_j)
    compute distance $d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$
    if d is < than the current min, then min = d
return min

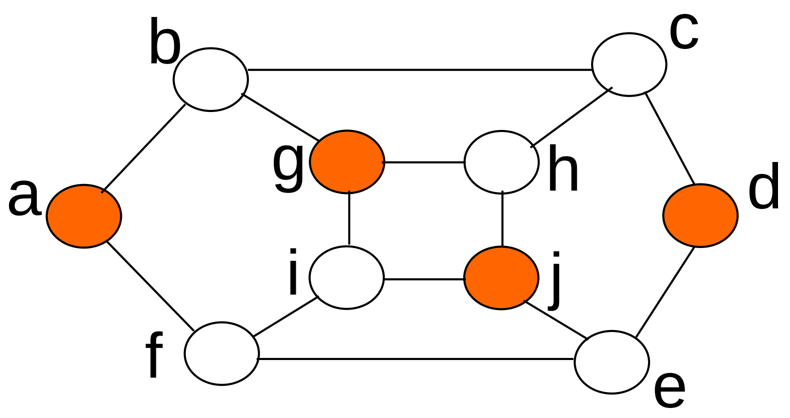
RT = $\Theta(n^2)$

Polynomial Time, $O(n^k)$ where k - constant

Example problem

Given a graph G with n nodes, find whether G has an ***independent set*** *of size k.*

A set S of nodes in G, $S \subseteq V$, is independent if no two nodes in S are joined by an edge.



S = {a, g, j, d} is an independent set of size 4

Polynomial Time, $O(n^k)$ where k - constant

Solution

- Take all the groups of k nodes and check if any group forms an independent set
- The number of groups of k nodes is $\binom{n}{k} = \Theta(n^k)$
- To check if a set of k nodes forms an independent set takes $\binom{k}{2} = \Theta(k^2)$

- Since k is constant, the total RT = $\Theta(n^k)$

Exponential Time, $\Theta(c^n)$ where c - constant

<u>Example problem</u>

Given a graph G with n nodes, find an independent set of maximum size

<u>Solution</u>

- Take all groups of nodes, and check if it forms an independent set
- Return the independent set of maximum size

<u>RT analysis</u>

- The number of groups is 2^n
- To check that a set of k nodes ($1 \leq k \leq n$) is independent set takes $\binom{k}{2}$ $= \Theta(k^2)$

- The total RT = $\Theta(2^n n^2)$

Factorial Time, $\Theta(n!)$

Example problem - Traveling Salesman Problem

Given a set of n cities, with distances between all pairs of cities, what is the shortest tour that visits all cities?

- NP – complete problem

Solution: enumerate all possible tours, then chose the shortest one

- The total RT = $\Theta(n!)$