Lecture 16 Notes

• **Goals for this week**
  – Graph basics
    • Types of graphs: Complete, Tree, Bipartite, …
    • Paths, Trails, Circuits, Cycles: Eulerian and Hamiltonian graphs
    • Chromatic number
  – Recurrences revisited

• **From last week**
  – Final exam review—Sunday June 8, 6pm-8pm
    • Request for room has been made. Room is still TBD.
    • Practice final exam questions – early Week 10
  – Week 9 and Week 10 quizzes off-line
    • (e.g., 12-hour window to take a quiz using WeBWorK)
  – “Things to Know” – end of Week 9, end of Week 10
Useful: logs, polynomials, exponentials

• Lecture 15: For all constants $c > 0$, $a > 1$, and for all monotonically growing $f(n)$: $(f(n))^c \in O(a^{f(n)})$

• $f(n) = n \Rightarrow \forall c > 0, a > 1, n^c \in O(a^n)$

Any exponential in $n$ grows faster than any polynomial in $n$

• $f(n) = \log_a n \Rightarrow \forall c > 0, a > 1, (\log_a n)^c \in O(a^{\log_a n}) = O(n)$

Any polynomial in $\log n$ grows slower than $n^{c'}$, $c'>0$

$log n \in O(n)$
$log^3 n$ [i.e., $(\log n)^3] \in O(n)$
Useful: “Unrolling” of Recurrences

Lecture 14: “first line of attack if no convenient theorem available…”

\[ T(n) = 4T(n/2) + \Theta(n) \]

- \( T(n) \leq 4T(n/2) + cn \)
  - \( \leq 4 [4T(n/4) + cn/2] + cn \)
    - \( = 16 T(n/4) + (1 + 2)cn \)
  - \( \leq 16 [4T(n/8) + cn/4] + (1 + 2)cn \)
    - \( = 64 T(n/8) + (1 + 2 + 4)cn \)
  - \( \ldots \)
  - \( \leq 4^k T(n/2^k) + (1 + 2 + 4 + \ldots + 2^{k-1})cn \)

For \( k = \log_2 n \):

\[ T(n) \leq n^2 T(1) + cn^2 = O(n^2) \]
Useful: “Master Theorem” for D/Q Recurrences

Theorem 8, GT-47

\[ T(n) \leq a \cdot T(n/b) + O(n^d) \]

- **Mergesort** \( T(n) = 2T(n/2) + \Theta(n) \)
  - \( a = 2, b = 2, d = 1 \)

- **Matrix Multiply** \( T(n) = 8T(n/2) + \Theta(n^2) \)
  - \( a = 8, b = 2, d = 2 \)
Recurrence \[ T(n) \leq a \cdot T(n/b) + O(n^d) \]

- **a**: #subproblems in “divide”
- **b**: size decrease factor

Work done / “paid for” at \( k^{th} \) level = \( a^k \times O(n/b^k)^d = O(n^d) \times (a / b^d)^k \)
Recurrence: \( T(n) \leq a \cdot T(n/b) + O(n^d) \)

1. if \( a < b^d \) \( T(n) = O(n^d) \)
2. if \( a = b^d \) \( T(n) = O(n^d \log n) \)
3. if \( a > b^d \) \( T(n) = O(n^{\log_b a}) \)

work done at \( k^{th} \) level is \( a^k \cdot O(n/b^k)^d = O(n^d) \cdot (a/b^d)^k \)
Useful: Change of Variables

\[ T(n) = 2T(\sqrt{n}) + 1 \]

Let \( k = \sqrt{n} \)
\[ T(k^2) = 2T(k) + 1 \]

Let \( m = \log_2 k \)
\[ T(2^{2m}) = 2T(2^m) + 1 \]

Let \( T(2^m) = F(m) \)
\[ F(2m) = 2F(m) + 1 \]
\[ F(m) = 2F(m/2) + 1 \]

== Master Theorem: \( a = 2, b = 2, d = 0 \)
Case (3): \( F(m) = O(m^{\log_2 2}) = O(m) \)

So, \( T(2^m) = O(m) \)

And, \( m = \log_2 k \Rightarrow T(k) = O(\log k) \)
Useful: Draw the Recursion Tree

\[ T(n) = T(n/3) + T(2n/3) + n \]

Lower-bounding the number of “full” levels, and upper-bounding the total number of levels \( \Rightarrow \) can get \( T(n) \in \Theta(n \log n) \)
Useful: Inhomogeneous Linear Recurrences

• General solution = general solution to homogeneous relation plus any particular solution to inhomogeneous relation

• Example: \( a_n = 3a_{n-1} + 2 \) or \( a_n = ca_{n-1} + f(n) \) (**)

• Homogeneous relation: delete the \( f(n) \) part \( \Rightarrow a_n = ca_{n-1} \)

  has general solution \( a_n = Ac^n \)

• If \( a^*_n \) is a particular solution of (**), i.e., \( a^*_n = ca^*_{n-1} + f(n) \), then \( a_n = Ac^n + a^*_n \) satisfies (**):

  \[
  a_n = Ac^n + a^*_n \\
  = (c \cdot Ac^{n-1}) + (ca^*_{n-1} + f(n)) \\
  = c(Ac^{n-1} + a^*_{n-1}) + f(n) \\
  = ca_{n-1} + f(n)
  \]
Useful: Inhomogeneous Linear Recurrences

- Example: \( a_n = c a_{n-1} + f(n) \)

**Useful special case:** \( c = 1 \Rightarrow a_n = a_{n-1} + f(n) \)

- Iterate (recall “unrolling” / “substitution”):
  \[
  a_1 = a_0 + f(1) \\
  a_2 = a_1 + f(2) = (a_0 + f(1)) + f(2) \\
  a_3 = a_2 + f(3) = (a_0 + f(1) + f(2)) + f(3) \\
  \ldots \\
  a_n = a_0 + f(1) + f(2) + f(3) + \ldots + f(n) \\
  = a_0 + \sum_{k=1}^{n} f(k)
  \]
Inhomogeneous Linear Recurrences

- \( a_n = a_{n-1} + n \)

- Number of regions created by \( n \) mutually intersecting lines in the plane

- \( a_n = 1 + (1 + 2 + 3 + \ldots + n) \)

\[
a_n = 1 + [n(n+1)/2]
\]
Useful: Inhomogeneous Linear Recurrences

- $a_n = ca_{n-1} + f(n)$  Other case: $c \neq 1$

<table>
<thead>
<tr>
<th>$f(n)$</th>
<th>particular solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d$ (a constant)</td>
<td>$B$ (a constant t.b.d.)</td>
</tr>
<tr>
<td>$dn$</td>
<td>$B_1n + B_0$</td>
</tr>
<tr>
<td>$dn^2$</td>
<td>$B_2n^2 + B_1n + B_0$</td>
</tr>
<tr>
<td>$d^n$</td>
<td>$Bd^n$</td>
</tr>
</tbody>
</table>

Example: $a_n = 2a_{n-1} + 1$ with $a_1 = 1$  (Tower of Hanoi)

- General soln to homogeneous relation $a_n = 2a_{n-1}$: $a_0^n = A2^n$
- Particular soln to inhomogeneous relation: set $a^*_n = B$
- $B = a^*_n = 2a^*_{n-1} + 1 = 2B + 1 \Rightarrow B = -1$  find particular solution first!
- $a^*_n = -1 \Rightarrow$ general solution $a^n = a^0_n + a^*_n = A2^n - 1$
- I.C.: $1 = a_1 = A2^1 - 1 \Rightarrow 2 = 2A \Rightarrow A = 1$: finally, $a_n = 2^n - 1$
### Master Theorem, Extended a Bit …

<table>
<thead>
<tr>
<th>$a, b$</th>
<th>$f(n)$ ($=n^d$)</th>
<th>$T(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a = 1$</td>
<td>$c$</td>
<td>$\log_d n$</td>
</tr>
<tr>
<td>$a = b$</td>
<td>$c$</td>
<td>$A_1n + A_2$</td>
</tr>
<tr>
<td>$a &lt; b$</td>
<td>$cn$</td>
<td>$A_1n$</td>
</tr>
<tr>
<td>$a = b$</td>
<td>$cn$</td>
<td>$O(n \cdot \log_b n)$</td>
</tr>
<tr>
<td>$a &gt; b$</td>
<td>$cn$</td>
<td>$O(n^{\log_b a})$</td>
</tr>
</tbody>
</table>

(From Tucker’s *Applied Combinatorics* text)
The MaxMin Problem

• Given a set \( S \) of \( n \) numbers, use divide-and-conquer to find the maximum element \textit{and} the minimum element in \( S \)

  Split \( S \) into two subsets of size \( n/2 \) each
  Recursively find the max and min of each subset
  Compare the two max’s, compare the two min’s

• Recurrence: \( T(n) = 2T(n/2) + 2 \)

  This is the “\( a = b \)” case in the previous slide
The MaxMin Problem

- **Recurrence:** \( T(n) = 2T(n/2) + 2 \)

\[
a_n = A_1 n + A_2
\]

\[
A_1 n + A_2 = a_n
\]

\[
= 2a_{n/2} + 2 = 2(A_1 n/2 + A_2) + 2
\]

\[
= A_1 n + 2A_2 + 2 \quad \Rightarrow A_2 = -2
\]

I.C.: \( a_2 = 1 \)

\[
\Rightarrow 1 = a_2 = A_1 2 - 2
\]

\[
\Rightarrow A_1 = \frac{3}{2}
\]

\[
\Rightarrow \text{Finally: } a_n = \left(\frac{3}{2}\right)n - 2
\]
Whew.
Graphs
(GT Sections 1, 2)

- A graph is a model of relationships between pairs of objects.
  - Objects: nodes / vertices
  - Relationships: edges / links / arrows

| Simple graph          | (V,E)                          | • Undirected  
|                      |                               | • (E ⊆ P₂(V))  
|                      |                               | • No self loops  
|                      |                               | • No edge labels or weights  
|                      |                               | GT-2  

| Graph                 | (V,E, φ)                        | • Undirected  
|                      |                               | • (E ⊆ P₂(V))  
|                      |                               | • No self loops  
|                      |                               | • Has edge labels  
|                      |                               | GT-3  

| Simple graph with loops | (V,E, φ)                          | • Undirected  
|                      |                               | • (E ⊆ P₂(V))  
|                      |                               | • May have self loops  
|                      |                               | • Has edge labels  
|                      |                               | GT-4  

| Digraph               | (V,E, φ)                        | • Directed  
|                      |                               | • (E ⊆ V × V)  
|                      |                               | • May have self loops  
|                      |                               | • Has edge labels  
|                      |                               | GT-15  

Representing graphs

- **Adjacency list**: for each vertex, list neighbors
- **Adjacency matrix**: $A_{ij} = 1$ iff vertices $i$ and $j$ are adjacent

*Graph diagram*

- **List**:  
  1 : \{2, 4\}  
  2 : \{1, 3, 4\}  
  3 : \{2\}  
  4 : \{1, 2\}

- **Matrix**:  
  \[
  \begin{bmatrix}
  0 & 1 & 0 & 1 \\
  1 & 0 & 1 & 1 \\
  0 & 1 & 0 & 0 \\
  1 & 1 & 0 & 0 \\
  \end{bmatrix}
  \]

  - $A_{14} = A_{41} = 1$
  - $A_{23} = A_{32} = 1$

  *(symmetric adjacency matrix for an undirected graph...)*
Graph Properties

• What is the maximum number of edges in a simple graph with n vertices?
  A. n
  B. \( n^2 \)
  C. \( 2^n \)
  D. \( C(n,2) \)
  E. None of the above.

A complete graph on n vertices is one with \( C(n,2) = \frac{n(n-1)}{2} \) edges.
Graph Properties

• If we start with n vertices and, for each pair of distinct vertices decide randomly (by flipping a fair coin) whether to place an edge between them, what is the expected number of edges in the graph?

   A. n/2
   B. n^2 – 1
   C. 2^{n-1}
   D. n(n-1)/4
   E. None of the above.

• Recall from homework: This is a random graph – specifically, a graph in G(n, 0.5).
The degree of vertex $v$ in graph $G$ is the number of edges incident to $v$.

The degree sequence of a graph $G$ is the list of all degrees of vertices in $G$, sorted in non-decreasing order.

What is the degree sequence of this graph?

A. 1, 2, 3, 4
B. 1, 2, 2, 3
C. 2, 3, 1, 2
D. 1, 2, 2
E. None of the above
Properties of Degree Sequences

• Theorem: If $G = (V,E)$ is a simple graph

$$\sum_{v \in V} d(v) = 2 \times |E|$$

• Every edge has two “endpoints”
• Sometimes called “handshake lemma”

• Consequences
  – The sum of vertex degrees is even
  – The number of odd-degree vertices is even
Isomorphic graphs

- Informally, two graphs are *isomorphic* if you can transform the picture of one into the other by “sliding vertices around and bending, stretching, and contracting edges as needed” and possibly relabeling the vertices.

Which pair of graphs is isomorphic?

A. I and III
B. II and IV
C. III and IV
D. I and II
E. None of the above.
Properties of isomorphic graphs

• If two graphs are isomorphic, then they have the same
  – numbers of vertices and number of edges
  – degree sequences
  – numbers of connected components  (don’t know about these yet …)
Special kinds of graphs

- **Complete** (simple) graphs: all possible edges

[Images of complete graphs with different numbers of vertices and edges]

http://en.wikipedia.org/wiki/File:3-simplex_graph.svg
Special kinds of graphs

- **Bipartite graphs**
  - Two “kinds” of vertices $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \emptyset$
  - Each edge is between a vertex of one kind and a vertex of the other kind: $e \in E$ has form $\{u \in V_1, v \in V_2\}$
Special kinds of graphs

- **Trees**
  - Connected
  - $|V| - 1$ edges
  - No cycles

(any two of these properties implies the third !)
Coloring a graph

- A proper coloring of a simple graph $G = (V,E)$ is a function $\lambda : V \rightarrow C$, where $C$ is a set of colors, such that $\{u,v\} \in E \rightarrow \lambda(u) \neq \lambda(v)$.

- There is a proper coloring of a complete graph on 3 vertices using:
  - A. 1 color
  - B. 2 colors (but there aren’t any with 1 color)
  - C. 3 colors (but there aren’t any with 1 or 2 colors)
  - D. 4 colors (but there aren’t any with 1 or 2 or 3 colors)
  - E. None of the above.
Coloring a graph

• A proper coloring of a simple graph $G = (V,E)$ is a function $\lambda : V \rightarrow C$, where $C$ is a set of colors, such that $\{u,v\} \in E \rightarrow \lambda(u) \neq \lambda(v)$

• There is a proper coloring of any bipartite graph using:
  A. 1 color
  B. 2 colors (but there aren’t any with 1 color)
  C. 3 colors (but there aren’t any with 1 or 2 colors)
  D. 4 colors (but there aren’t any with 1 or 2 or 3 colors)
  E. None of the above.
Coloring problem and chromatic numbers

- The **coloring problem**: for any \( c > 2 \), devise an algorithm whose input can be any simple graph and whose output is the answer to the question “can the graph be properly colored with \( c \) colors?”

- This is an **NP-complete** problem: no known fast algorithm to come up with solution; but easy to check if candidate solution works.

- The **chromatic number** of a graph \( G \), \( \chi(G) \), is the least number of colors needed to properly color \( G \).

- In the previous example, we saw:
  \[
  \chi(\text{Complete graph, } n \text{ vertices}) = n, \\
  \chi(\text{Bipartite graph}) = 2
  \]

- The **Four Color Theorem** says that every planar MAP can be properly colored with four colors. Does this mean that \( \chi(\text{any planar graph}) \leq 4 \) ?
  - **Planar Graph**: can be drawn on a plane such that edges intersect only at endpoints
Problems 16

• P16.1 Prove that for any \( n \geq 2 \), \( \frac{1}{2^2} + \frac{1}{3^2} + \cdots + \frac{1}{n^2} < 1 \).

• P16.2 Two players take turns removing stones from a pile. Whoever removes the last stone wins. They can remove 1 to \( n \) stones each turn. Find the winning strategy for each player.

• P16.3 You play a game against Bob. You have a very big urn, and at the start of the game, there are twenty balls, two each numbered one through ten. At each turn, you throw out one ball, and Bob dumps in any (finite) number of his choice of balls with smaller (positive integer) labels. For example, if you throw out a ball labeled ten, Bob could dump in a million balls labeled four. But if you throw out a ball labeled one, then Bob can’t do anything. You win if you can empty the urn. Can you win? What is your strategy?
Problems 16

• P16.4 Each year the country of Philatelia introduces a new stamp, for a denomination (in cents) that cannot be achieved by any combination of older stamps. Show that at some point the country will be forced to introduce a 1-cent stamp.

• P16.5 On an $n \times n$ board there are $n$ squares, $n - 1$ of which are infected. Each second, any square that is adjacent to at least two infected squares becomes infected. Can the entire board be infected?