CMPSC112
Lecture 25: Trees
Prof. John Wenskovitch
03/28/2016
Panel Presentation
April 5, 2016 from 11:00 am - 12:15 pm in Alden 101

Liberal Arts + ? = Awesome Career in Computing

Benjamin Cammarano
Microsoft, Class of 1992

Mike Capsambelis
Google, Class of 1994

Anna Griffiths
Discovery Machine, Class of 1991
Cupper Scholars Program
(April 8 Deadline)

Are you interested in furthering your knowledge of computer science by conducting collaborative research with a faculty member at Allegheny College? If yes, then please consider applying for a fellowship supported by the Cupper Scholars program! Recognizing the noteworthy contributions of the late Dr. Robert D. Cupper, the founder of the Department of Computer Science at Allegheny College and a ground-breaking and innovative teacher and scholar in the field of computer science, this program provides students with mentoring, learning, and research opportunities. To find out more about the Cupper Scholars Program, please talk with the Chair of the Department of Computer Science. To see Professor Cupper teaching in Alden Hall, you can watch several segments in the above video!

Requirements
- A minimum overall GPA of 3.0
- A final copy of your resume and a cover letter
- A declared major or minor in computer science
- Printed transcripts (a version from WebAdvisor is acceptable)
- A personal interest statement that explains what type of research you would like to conduct and how it will benefit both the department and the discipline of computer science

Benefits Afforded to a Cupper Scholar
- Invitation to a scholar-only event during the academic year
- Mentoring from alumni who support the Cupper Scholars program
- A stipend of approximately $1200 to support your collaborative research project
- Research and career advice from mentors during the spring and/or summer academic semesters
Office Hours Canceled April 8

<table>
<thead>
<tr>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
</tr>
</thead>
<tbody>
<tr>
<td>9am</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10am</td>
<td>10 – 11 Office Hours (15min slots)</td>
<td>10 – 11:30 Office Hours (15min slots)</td>
<td>10 – 11 Office Hours (15min slots)</td>
<td>10 – 5p Research/Medical</td>
</tr>
<tr>
<td>11am</td>
<td>11 – 11:50 CMPSC112</td>
<td>11:30 – 12:30p Lunch</td>
<td>11 – 11:50 CMPSC112</td>
<td>11 – 11:50 CMPSC112</td>
</tr>
<tr>
<td>12pm</td>
<td>12p – 1p Lunch</td>
<td>12p – 1p Lunch</td>
<td>12p – 1p Lunch</td>
<td>12p – 1p Lunch</td>
</tr>
<tr>
<td>1pm</td>
<td>1:30p – 2:20p CMPSC250</td>
<td>1:30p – 2:30p CMPSC600</td>
<td>1:30p – 2:20p CMPSC250</td>
<td>1:30p – 2:20p CMPSC250</td>
</tr>
<tr>
<td>2pm</td>
<td>2:30p – 5:30p Office Hours (15min slots)</td>
<td>2:30p – 5p G2 Work and Skype Meeting</td>
<td>2:30p – 4:20p CMPSC112 Lab</td>
<td>2:30p – 4:20p CMPSC250 Lab</td>
</tr>
<tr>
<td>3pm</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4pm</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5pm</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6pm</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Last Time

• Hashtables
  – Array-backed constant-time data structure
  – Hash function maps key to array index
  – How do we handle two data items hashing into the same location?
    • Separate Chaining
    • Linear Probing
Trees

• A data type that stores nodes hierarchically.
Definitions

• A **tree** is a set of nodes storing elements such that the nodes have a **parent-child** relationship that satisfies the following:
  – If the tree is nonempty, it has a special node, called the **root**, that has no parent.
  – Each node different from the root has a unique **parent** node, and every node with a parent \( w \) is a **child** of \( w \).

• Two nodes that are children of the same parent are **siblings**.

• A node is **external** (or a **leaf**) if it has no children, and **internal** if it does have children.
More Definitions

- A tree is **ordered** if there is a meaningful order among the children of each node (some node is first, some other node is second, etc).
Even More Definitions

• An **edge** of a tree is a pair of nodes \((u,v)\) such that \(u\) is the parent of \(v\), or \(v\) is the parent of \(u\).

• A **path** is a sequence of nodes such that any two consecutive nodes in the sequence form an edge.

• A **subtree** rooted at node \(v\) is the tree consisting of all the **descendants** of \(v\) in the tree, including \(v\) itself.

• The **height** of a tree is the maximum distance from the root to a leaf node. The distance from a node to the root is the **depth** of that node.

• All nodes at the same depth are said to be on the same **level** of the tree.
More and More Definitions

• A binary tree is an ordered tree with the following properties:
  – Every node has at most two children.
  – Each child node is labeled as being either a left child or a right child.
  – A left child precedes a right child in the sibling order.

• A binary tree is called proper or full if each node has either zero or two children. A binary tree is improper if it is not proper.

• The left child is the root of the left subtree; the right child is the root of the right subtree.
Why Bother with Trees?
Any Questions?