PageRank Algorithm and Quiz Review
Announcement

- March 3, guest lecturer Ross Dimassimo with the help of William Garnes III
- March 3, Quiz 4
- Bonus 2 Project: Python Art for T-shirt, due today
Quiz 4 Review
Quiz 4: Sorting

- Selection Sort
- Insertion Sort
- Merge Sort
Review 1: Selection Sort
Algorithm for Selection Sort

Input: a list, *Unsorted*, of unordered items
Initialization: set *Sorted* to empty

while (items remain in *Unsorted*)
    find the smallest item in *Unsorted*
    put that item on the end of *Sorted*

Output: the finished list *Sorted*
### Selection Sort Example

<table>
<thead>
<tr>
<th>Unsorted</th>
<th>Min</th>
<th>Sorted</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 2 1 4 3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5 2 4 3</td>
<td>2</td>
<td>1 2</td>
</tr>
<tr>
<td>5 4 3</td>
<td>3</td>
<td>1 2 3</td>
</tr>
<tr>
<td>5 4</td>
<td>4</td>
<td>1 2 3 4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>1 2 3 4 5</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>1 2 3 4 5</td>
</tr>
</tbody>
</table>
# Selection Sort Exercise

<table>
<thead>
<tr>
<th>Unsorted List</th>
<th>Min</th>
<th>Sorted list</th>
</tr>
</thead>
<tbody>
<tr>
<td>25, 8, 42, 16, 77</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Selection Sort Exercise

<table>
<thead>
<tr>
<th>Unsorted List</th>
<th>Min</th>
<th>Sorted list</th>
</tr>
</thead>
<tbody>
<tr>
<td>25, 8, 42, 16, 77</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>25, 42, 16, 77</td>
<td>16</td>
<td>8, 16</td>
</tr>
<tr>
<td>25, 42, 77</td>
<td>25</td>
<td>8, 16, 25</td>
</tr>
<tr>
<td>42, 77</td>
<td>42</td>
<td>8, 16, 25, 42</td>
</tr>
<tr>
<td>77</td>
<td>77</td>
<td>8, 16, 25, 42, 77</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>8, 16, 25, 42, 77</td>
</tr>
</tbody>
</table>
Review 2: Insertion Sort
<table>
<thead>
<tr>
<th>Unsorted</th>
<th>Top Value</th>
<th>Insert After</th>
<th>Sorted</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 2 1 4 3</td>
<td>5</td>
<td>front</td>
<td>5</td>
</tr>
<tr>
<td>2 1 4 3</td>
<td>2</td>
<td>front</td>
<td>2 5</td>
</tr>
<tr>
<td>1 4 3</td>
<td>1</td>
<td>front</td>
<td>1 2 5</td>
</tr>
<tr>
<td>4 3</td>
<td>4</td>
<td>2</td>
<td>1 2 4 5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1 2 3 4 5</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1 2 3 4 5</td>
</tr>
</tbody>
</table>
## Insertion Sort Exercise

<table>
<thead>
<tr>
<th>Unsorted List</th>
<th>Top Value</th>
<th>Insert After</th>
<th>Sorted list</th>
</tr>
</thead>
<tbody>
<tr>
<td>25, 8, 42, 16, 77</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
### Insertion Sort Exercise

<table>
<thead>
<tr>
<th>Unsorted List</th>
<th>Top Value</th>
<th>Insert After</th>
<th>Sorted list</th>
</tr>
</thead>
<tbody>
<tr>
<td>25, 8, 42, 16, 77</td>
<td>25</td>
<td>front</td>
<td>25</td>
</tr>
<tr>
<td>8, 42, 16, 77</td>
<td>8</td>
<td>front</td>
<td>8, 25</td>
</tr>
<tr>
<td>42, 16, 77</td>
<td>42</td>
<td>25</td>
<td>8, 25, 42</td>
</tr>
<tr>
<td>16, 77</td>
<td>16</td>
<td>8</td>
<td>8, 16, 25, 42</td>
</tr>
<tr>
<td>77</td>
<td>77</td>
<td>42</td>
<td>8, 16, 25, 42, 77</td>
</tr>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>8, 16, 25, 42, 77</td>
</tr>
</tbody>
</table>
Review 3: Merge Sort
Key step: Merge 2 sorted List

5 6

1 3

1 3 5 6

2 4 7 8

1 3 5 6

(1, 5) (3,5) - -

1 2 3 4 5 6 7 8

(1, 2) (2,3) (3,4) (4,5) (5,7) (6,7) - -
Merge Sort: exercise

12 25 36 42

11 24 37 41
# Merge Sort: exercise

<table>
<thead>
<tr>
<th>12</th>
<th>25</th>
<th>36</th>
<th>42</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>24</td>
<td>37</td>
<td>41</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11</th>
<th>12</th>
<th>24</th>
<th>25</th>
<th>36</th>
<th>37</th>
<th>41</th>
<th>42</th>
</tr>
</thead>
<tbody>
<tr>
<td>(11, 12)</td>
<td>(12, 24)</td>
<td>(24, 25)</td>
<td>(25, 37)</td>
<td>(36, 37)</td>
<td>(37, 42)</td>
<td>(41, 42)</td>
<td>–</td>
</tr>
</tbody>
</table>
Quicksort
https://www.youtube.com/watch?v=aQiWF4E8flQ
http://me.dt.in.th/page/Quicksort/
Quicksort

1. partition the array into two parts around a pivot
2. quicksort those smaller arrays
3. concatenate the two sorted arrays end to end
PageRank
The Basics

Readings:
http://interestingwebs.blogspot.com/2009/05/simple-explain-of-google-pagerank.html
http://www.sirgroane.net/google-page-rank/
What is PageRank?

- How Google determines a page’s relevance or importance.
- "PageRank" or "PR": a term to indicate the popularity of a page.
- The PR is determined by the number of links from other pages on the World Wide Web that point to this page.
- PR is like a vote by other pages in terms of its importance
- More votes, more important
- PR of the voters are also important in the computation
- Higher PR of voter page means better PR for the voted page
PageRank in Google

- PR does not directly influence a web page's ranking in the search engine results.
- PR doesn’t determine which webpages are included in the search results when a search term is entered.
- The search results ranking is determined by the relevance of titles, keywords and phrases contained within those pages.
- When two web pages have the same relevance to a search term, PR will determine which page is displayed first in the search results.
- PR is very important for search engine optimization (SEO).
Check pagerank

Use a PR Checker

http://www.prchecker.info/check_page_rank.php
The Values of PageRank

- Each PR level (1 – 10) is progressively harder to reach.
- PR is believed to be calculated on a logarithmic scale.

<table>
<thead>
<tr>
<th>Toolbar PageRank (log base 10)</th>
<th>Real PageRank</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 - 10</td>
</tr>
<tr>
<td>1</td>
<td>100 - 1,000</td>
</tr>
<tr>
<td>2</td>
<td>1,000 - 10,000</td>
</tr>
<tr>
<td>3</td>
<td>10,000 - 100,000</td>
</tr>
<tr>
<td>4</td>
<td>and so on...</td>
</tr>
</tbody>
</table>

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Number of websites with different PR values

Credit: http://interestingwebs.blogspot.com/2009/05/simple-explain-of-google-pagerank.html
Classification based on PR

- 0 – 3: new webpages or those with very few back links
- 4 – 5: popular pages with a lot of back links from similar sites
- 6: exceptionally popular sites with hundreds of links from authority sites
- 7 – 10: usually media brands, big corporations, or government sites
- check out: cnn.com, whitehouse.gov, utah.edu
How PR is calculated?

- Backlink: a link pointing to a page
- PR of a page is roughly based on the quantity of backlinks and the RP of the pages providing the links (voter pages).
- Other factors: relevance of search words on the page, actual visits to the page also influence the PR.
- No specific details are known about these factors
- To prevent manipulation, spoofing and spamdexing
How PR is calculated?

- **Spoofing**: falsifying the origin of an internet communication (emails, webpages) in order to mislead the recipient
- **Spamdexing** (search engine spam, search engine poisoning, Black-Hat SEO, search spam or web spam): deliberate manipulation of search engine indexes
Main Factors that Influences PR

- Number, relevance and quality of backlinks (incoming links)
  - The more backlinks the better
  - The more relevant and better quality of backlinks, the better
Main Factors that Influences PR

- PR of voter pages (where backlinks coming from)

Credit: http://interestingwebs.blogspot.com/2009/05/simple-explain-of-google-pagerank.html
Main Factors that Influences PR

- Outbound links of voter edges (more outbound links, worse the PR)

Credit: http://interestingwebs.blogspot.com/2009/05/simple-explain-of-google-pagerank.html
Other Facts on PageRank (PR)

- Bad backlinks, content of webpage do not impact RP
- PR does not rank web sites as a whole, but is determined for each page individually
- PRs are computed permanently, update every few months
- Efficient internal onsite linking has an impact on PR
- No one knows for sure how PR is calculated now
- PR can decrease
- Site can be banned if it links to banned sites
PageRank Algorithm

Readings:
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
http://interestingwebs.blogspot.com/2009/05/simple-explain-of-google-pagerank.html
http://www.sirgroane.net/google-page-rank/
PageRank or PR(A) can be calculated using a simple iterative algorithm, and corresponds to the principal eigenvector of the normalized link matrix of the web.

--- The original Google PageRank Paper
The Formula

$$PR(A) = (1-d) + d \left( \frac{PR(T1)}{C(T1)} + \frac{PR(T2)}{C(T2)} + \ldots + \frac{PR(Tn)}{C(Tn)} \right)$$

- $PR(A)$ is the PageRank of page A
- $PR(Ti)$ is the PageRank of pages Ti which link to page A
- $C(Ti)$ is the number of outbound links on page Ti
- $d$ is a damping factor which can be set between 0 and 1, treat it as probability math magic, e.g. 0.85
- $\frac{PR(Ti)}{C(Ti)}$: share of vote from page Ti
**Principle**

PageRank can be calculated using a simple iterative algorithm and corresponds to the principal eigenvectors of the normalized link matrix of the web.

We can calculate a page’s PR without knowing the final value of the PR of the other pages.

Each time we run the computation, we get one step closer to the final value.
A simple Example

Guess the PR of the following 2 pages.
C(A) = 1, C(B) = 1 (# of outgoing links)

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Guess 1

d = 0.85, guess PR(A) = 1, and PR(B) = 1

PR(A) = (1–d) + d(PR(B)/1) = 0.15 + 0.85 * 1 = 1
PR(B) = (1–d) + d(PR(A)/1) = 0.15 + 0.85 * 1 = 1

The guessed numbers did not change! We got away with a lucky guess!
\(d = 0.85, \text{ guess } PR(B) = 0\)

Step 1:

\[
PR(A) = (1-d) + d(PR(B)/1) = 0.15 + 0.85 \times 0 = 0.15
\]

\[
PR(B) = (1-d) + d(PR(A)/1) = 0.15 + 0.85 \times 0.15 = 0.2775 \text{ #Use new } PR(A)
\]

Step 2:

\[
PR(A) = 0.15 + 0.85 \times 0.2775 = 0.385875
\]

\[
PR(B) = 0.15 + 0.85 \times 0.385875 = 0.47799375
\]

Step 3:

\[
PR(A) = 0.15 + 0.85 \times 0.47799375 = 0.5562946875
\]

\[
PR(B) = 0.15 + 0.85 \times 0.5562946875 = 0.622850484375
\]

The values for PR(A) and PR(B) will converge to 1.

PageRank Algorithm: Rough Idea

Start with some random guess, iteratively update the PR until convergence (things settle down).
PageRank Algorithm: Random Walk Version

PR assigns a value to each web page, denoting the “importance” of a page under two assumptions:

1. For some fixed probability $a$, a surfer at a web page jumps to a random web page with probability $a$ and goes to a linked web page with probability $1 - a$.
2. The importance of a web page $v$ is the expected sum of the importance of all the web pages $u$ that precede $v$.

MORE ON THIS LATER: Bonus Project
Principle

It does not matter where you start your guess, once the PR calculation settles down, the normalized probability distribution (the average PR for all pages) will be 1.0.
Example 2

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 2

Average PR: 1.000

Page A
1.49

Page B
0.78

Page C
1.58

Page D
0.15

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 3

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 3

Note: external sites here wasted their PR by not voting for anyone else!
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 4

Average PR: 1.000

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 5

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 6: Loop

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 7

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Our HOME page PR has increased!
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Our HOME page PR has increased!
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Increasing the internal links in your site can minimize the damage to your PR when you give away votes by linking to external site.

Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 10: Site Map

Guess who has the highest PR?
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 10: Site Map

Guess who has the highest PR?
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
Example 10: Site Map

Guess who has the highest PR?
Credit: http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm
More ideas on increasing PR

1. Be a Mega-Site: many pages with rich content and links back to the parent/home page, e.g. news.bbc.co.uk
2. Content is King!
3. Make it worthwhile for other pages to use your content/tools
4. Getting thousands of links from sites with small PR may worth more than 1 link from a single site with large PR
THANKS!

Any questions?

You can find me at
beiwang@sci.utah.edu

http://www.sci.utah.edu/~beiwang/teaching/cs1060.html
Credits

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash