

אלגוריתמים היוריסטיים ומקורבים ויישומם 65344
Heuristic and Approximation Algorithms and Applications
סמסטר ק', תשע"ט

Professor Eugene LEVNER

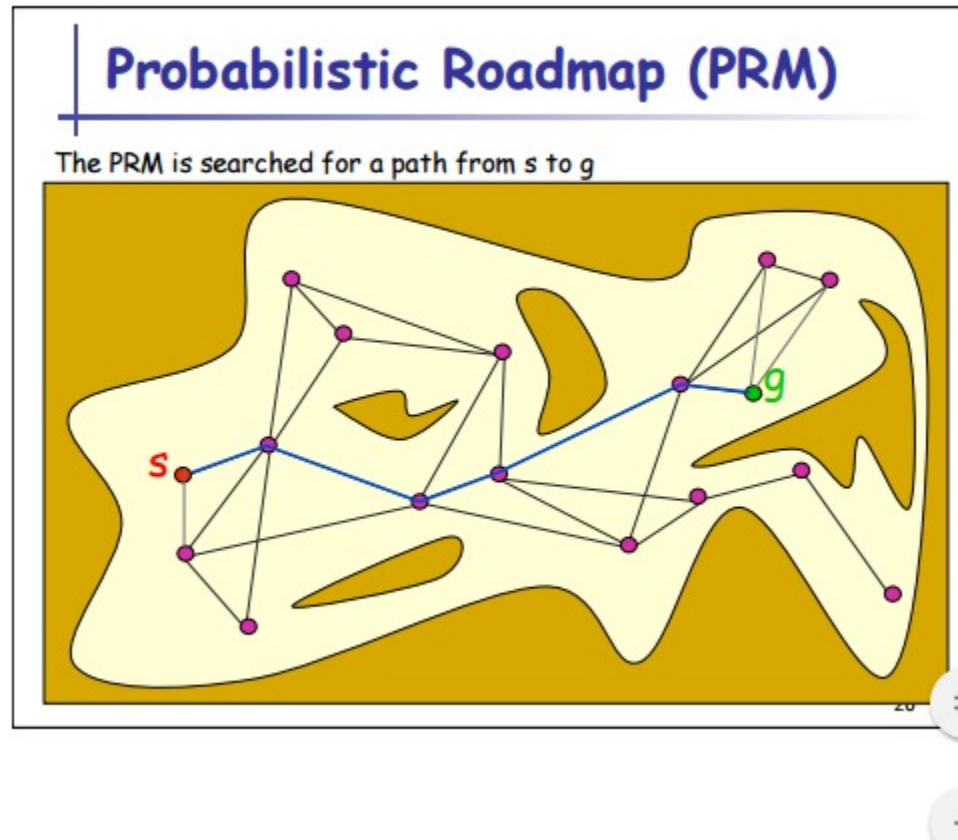
הרצאה 3

אלגוריתם PAGE-RANK

Summary of Lectures 1-2

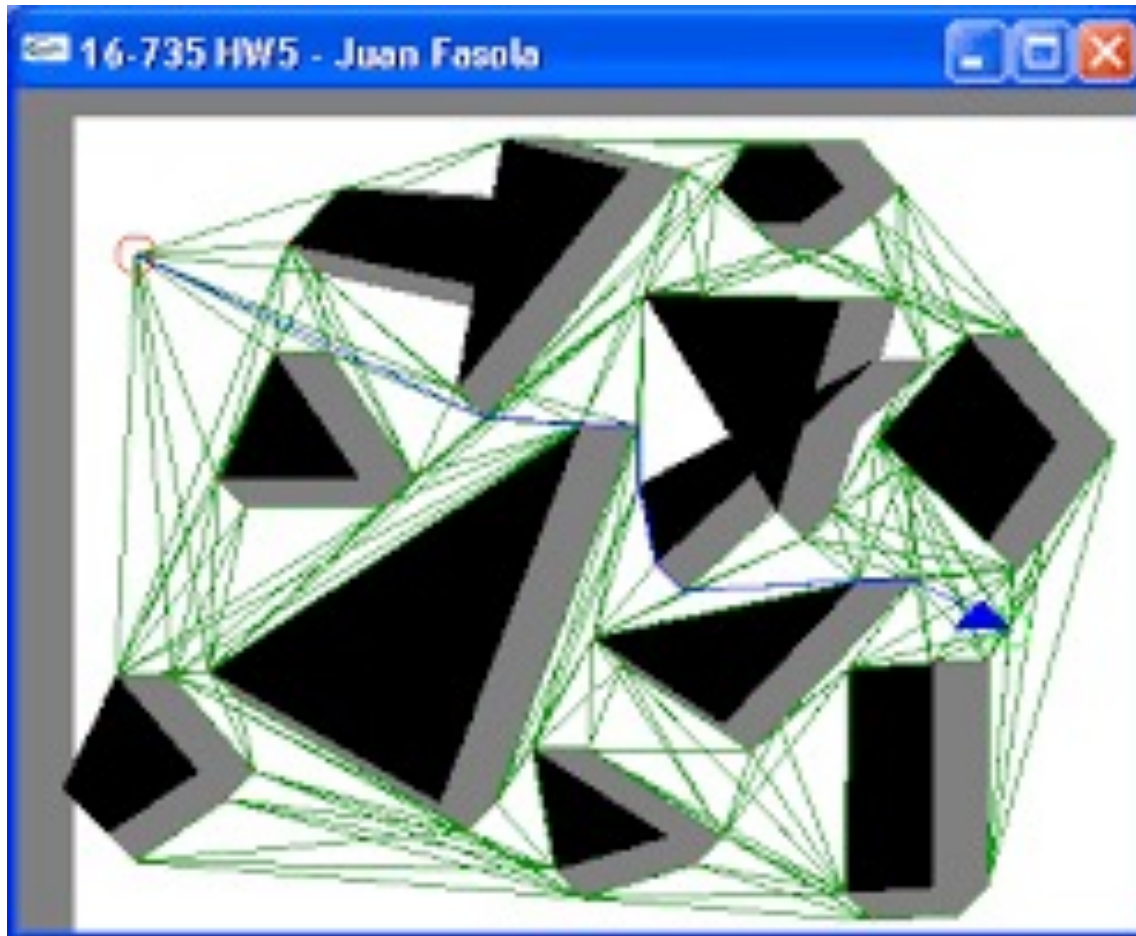
- 1. Advantages and disadvantages of PRM**
- 2. Advantages and disadvantages of A^***

PRM vs Dijkstra Algorithm

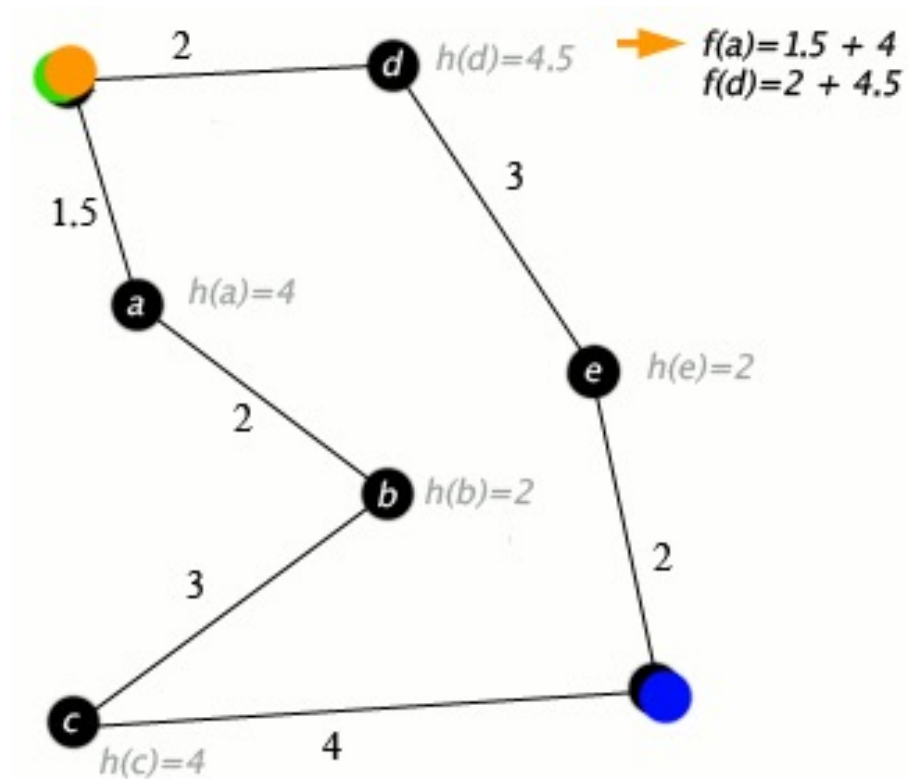


PRM. SPP with obstacles

(מכשולים)

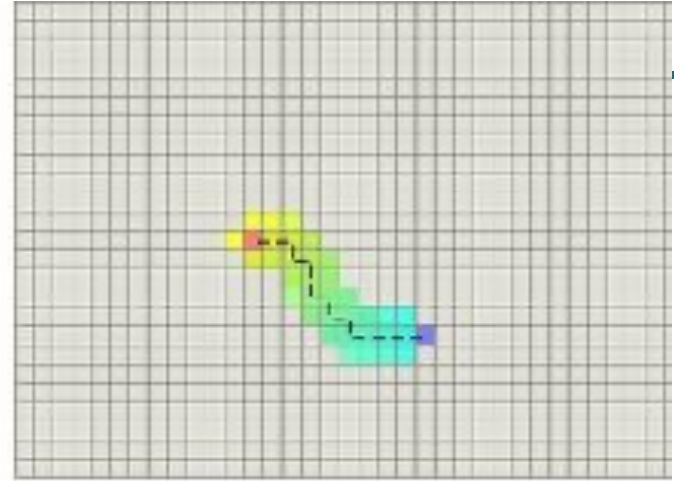
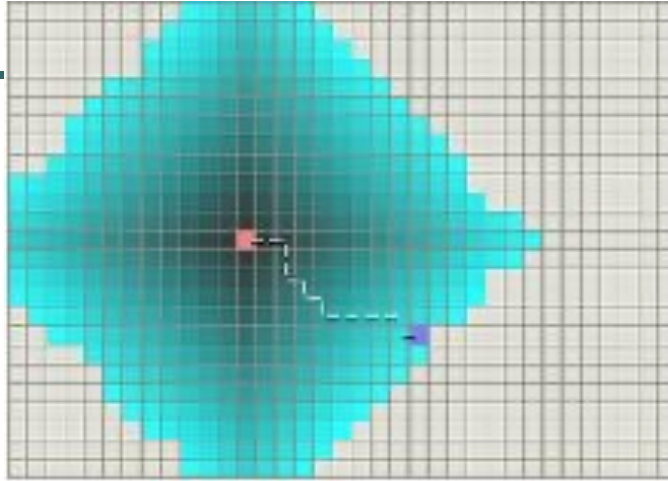


Class work. Is A^* an exact algorithm?

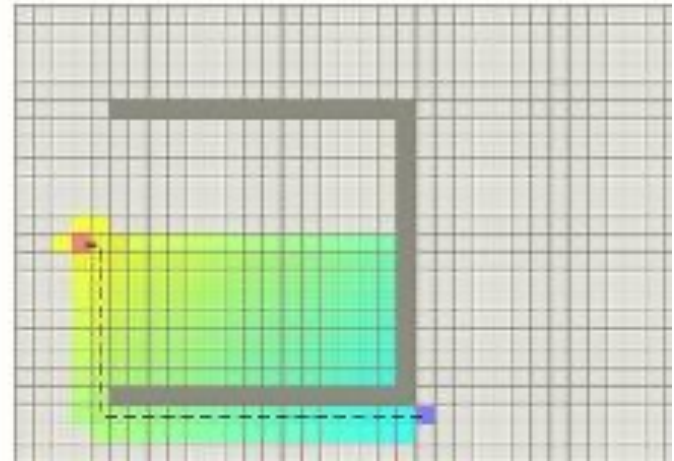
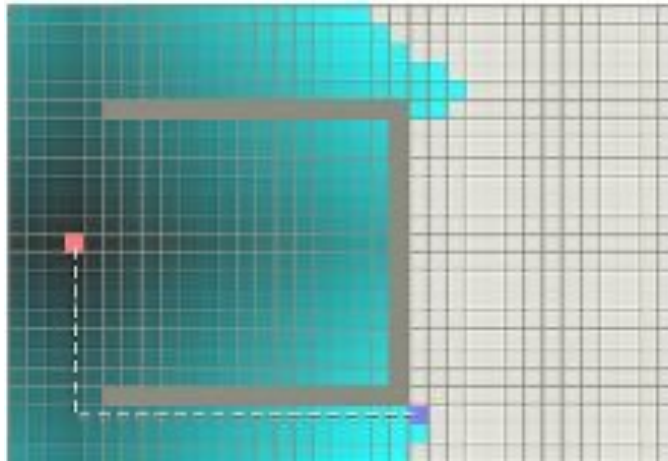


Dijkstra vs. A*

Without
obstacle



With
obstacle



Home work/Class work

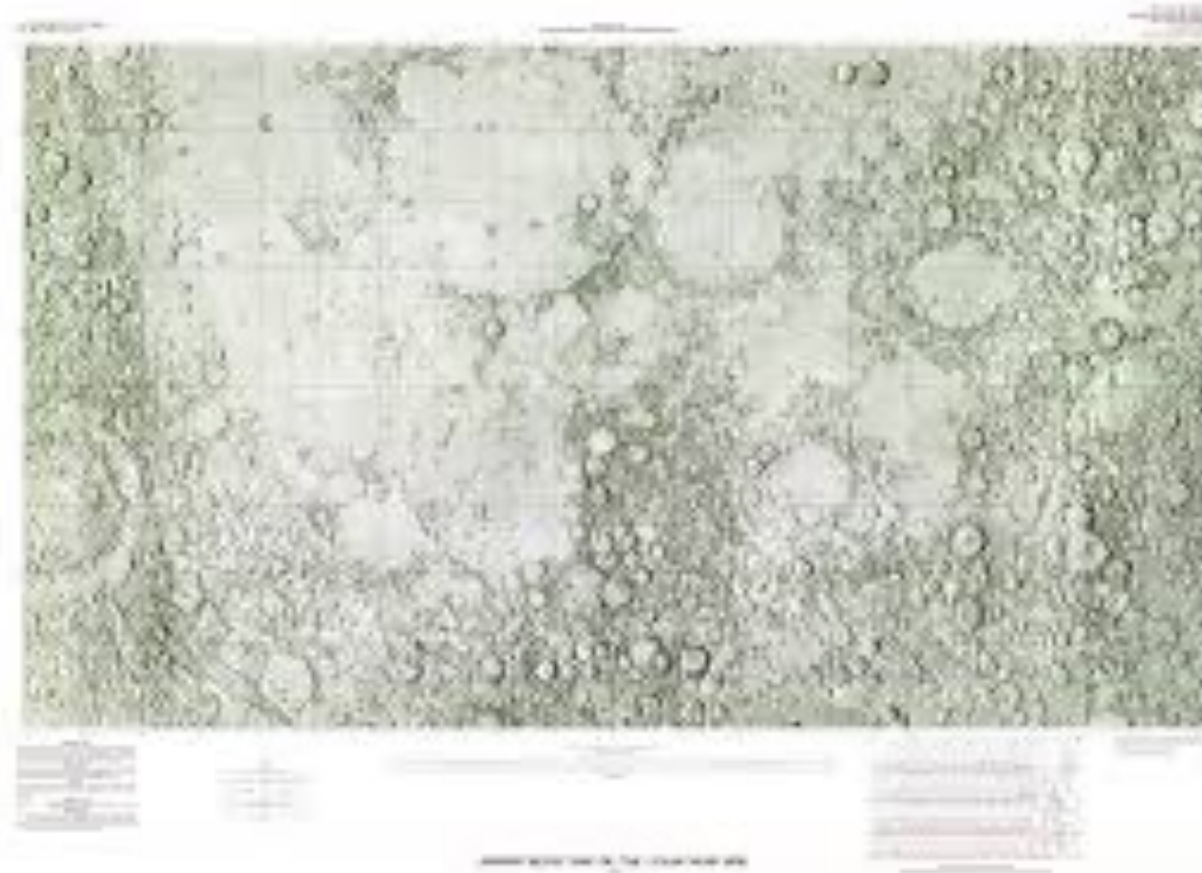
- To construct an example of a graph with 10 nodes and more than 12 arcs in which A* works better than Dijkstra.
- To construct an example of a graph with 10 nodes and at least 15 arcs in which A* works worse than Dijkstra.

Topics of Mini-Projects

6 נושאים של "mini-projects" ופרויקטים שנתיים

1. Robot on the Moon. Using Algorithm PRM for robot routing with random circular (עגולים) obstacles.
2. Robot on the Moon. Using Algorithm A* for robot routing with random circular (עגולים) obstacles
- 3-4. Two robots on the Moon Algorithms PRM (or A*) for robot routing with random circular (עגולים) obstacles and two robots.
5. Algorithm PageRank for solving the ranking problem with 40 sites.
Topic: Computer Algorithms
Computational comparison of three computational versions.
6. Multi-criteria ranking of 8 Israeli universities via TOPSIS.

Lunar maps



MAPPING MARSIAN CAVES IN 2D WITH A SMALL EXPLORATORY ROBOT

Contents of a project

- 1. Description of problem 10 points***
- 2. Description of the algorithm 20 points***
- 3. Solution of numerical examples
and comparison of algorithms 20 points***
- 5. Code on Python 20 points***
- 6 Solution of all home exercises 10 points***
- Defence of the project 20 points***

אלגוריתם PageRank •

(Crawler)(זחלן הרשת)

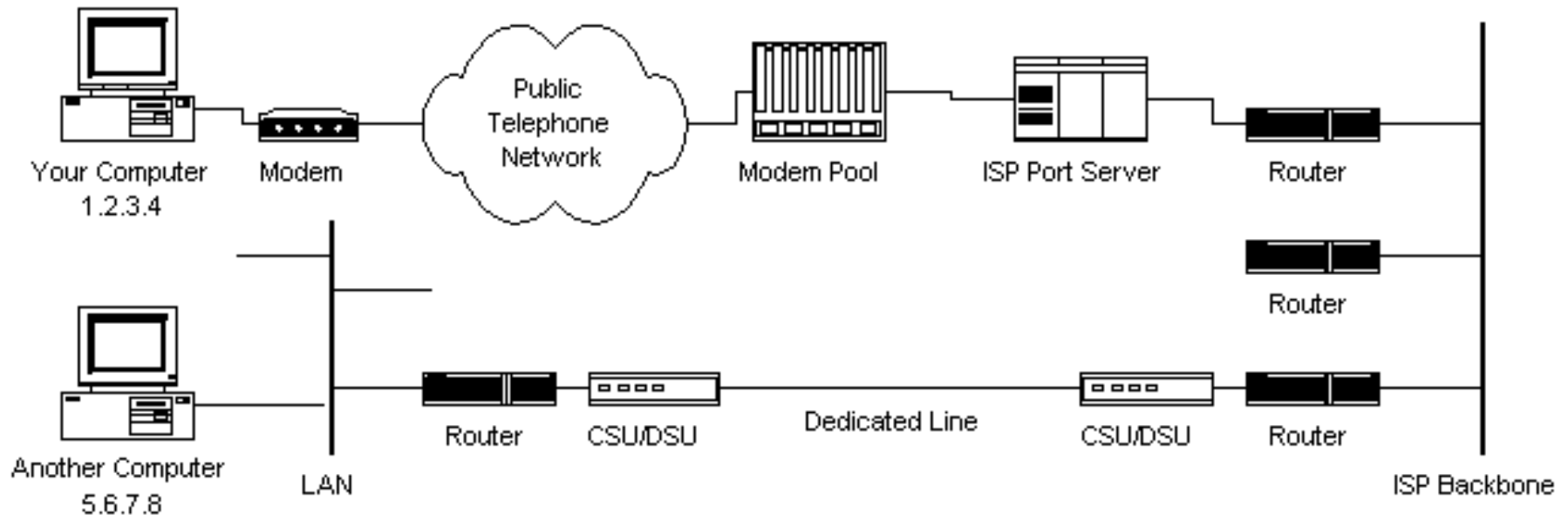
How does internet work?

How does PageRank work?

● כיצד פועל האינטרנט?

● כיצד פועל PageRank ?

Diagram 1



ISP =Internet Service Provider (ספק)

Backbone (עמוד שדרה) is made up of many large networks which interconnect with each other. They called **Network Service Providers** or **NSPs** router נָתָב

CSU/DSU (Channel Service Unit/Data Service Unit)

A brief history of Internet

- 1844: **Samuel Morse** transmits the first electric telegraph message משדר הודעת טלגרף
- 1876: **Alexander Graham Bell** develops ממציא the telephone.
- 1940: **George Stibitz** connects 2 computer by telephone in two cities
- 1969: The **ARPANET** computer network is launched, initially linking together four scientific institutions in California and Utah.
- 1971: **Ray Tomlinson** sends the first email, introducing the @ sign.
- 1973: **Bob Metcalfe** invents Ethernet, a way of linking computers and peripherals (like printers) on a local network.
- 1983: TCP/IP is officially adopted as the standard way in which Internet computers will communicate.

A brief history of Internet (2)

- 1989: **Tim Berners-Lee** invents the World Wide Web at CERN, the European physics laboratory in Switzerland.
- 1995: E-commerce properly begins when **Jeff Bezos** founds Amazon.com and **Pierre Omidyar** sets up eBay.
- 1998: **Larry Page** and **Sergey Brin** develop a search engine called BackRub that they quickly decided to rename Google.
- 2004: Harvard student **Mark Zuckerberg** creates Facebook, an easy-to-use website that connects people with their friends.

History

- The word Page Rank is a trademark of Google, and the PageRank process has been patented (U.S. Patent #6,285,999). However, the patent is assigned to Stanford University and not to Google. Google has exclusive license rights on the patent from Stanford University. The university received 1.8 million shares of Google in exchange for use of the patent; the shares were sold in 2005 for \$336 million.

The authors of PageRank

- Sergey Brin and Larry Page



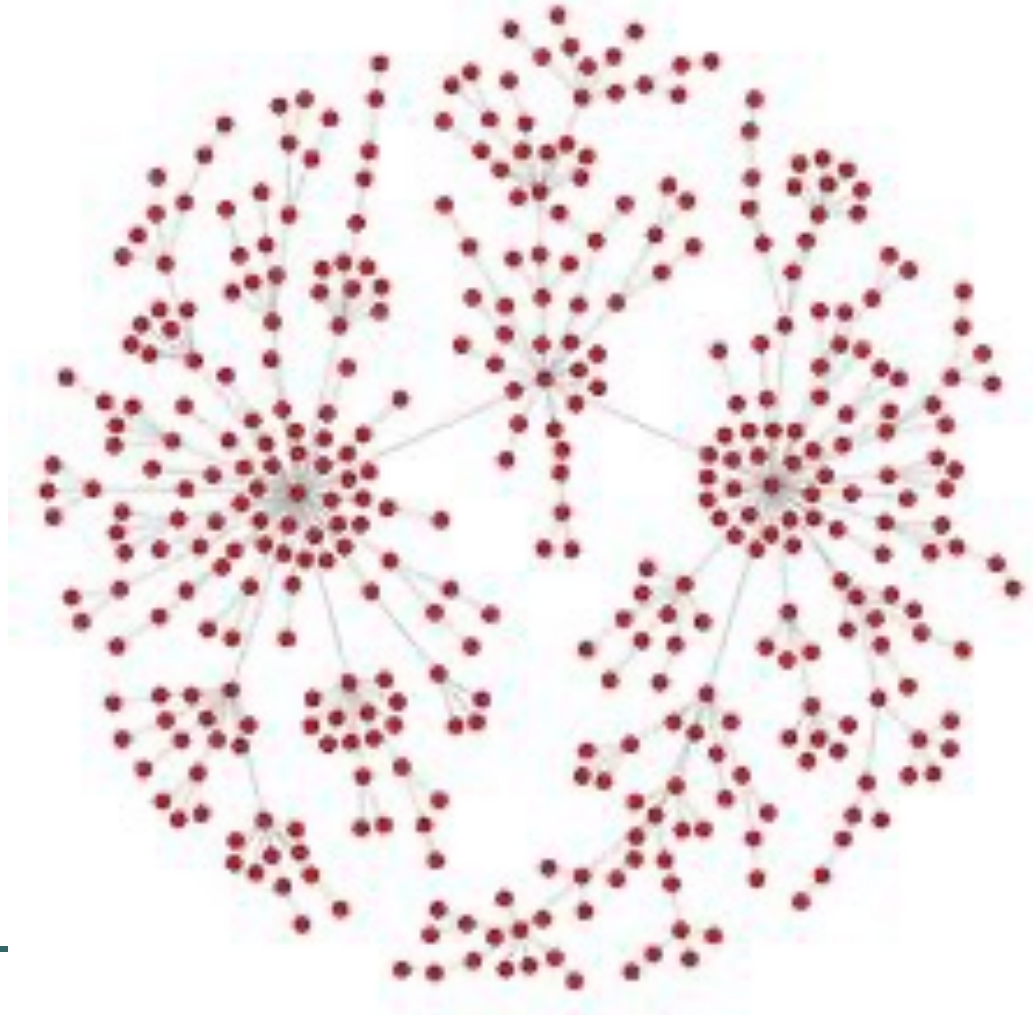
- **PageRank** is an algorithm used by Google Search to rank websites in their search engine results. PageRank was named after Larry Page.

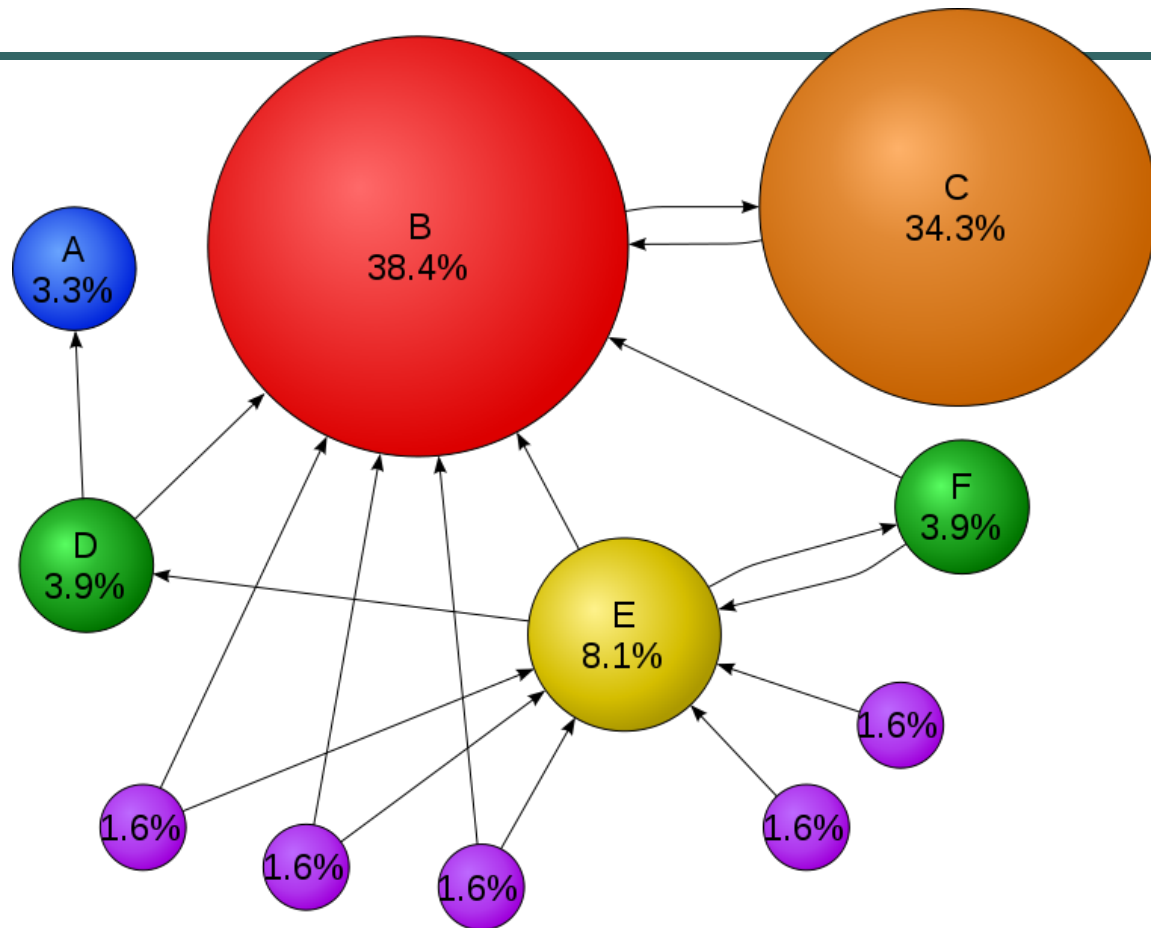
"פייג' ראנק" מסתמך על האופי הדמוקרטי הייחודי של הרשת ומשתמש במבנה הקישוריות העצום שלה כאינדיקציה לערכו של כל אתר. בעיקרון, **גוגל מפרש קישור מאתר A לאתר B, כהצבעה של A ל B**, אבל גוגל בוחן יותר מאשר את מספר ההצבעות שכל אתר מקבל, כלומר מספר קישורים שאתר מסוים מקבל; **בנוסף, גוגל מנתח את האתר המצביע**. הצבעות מאתר "חשוב" הן בעלות משקל גבוה יותר, ועוזרות לקדם אתרים אחרים."



"פייג' ראנק" מסתמך על האופי הדמוקרטי
הייחודי של הרשת ומשתמש במבנה הקישוריות
העצום שלה כאינדיקציה לערכו של כל אתר.
בעיקרון, גוגל מפרש קישור מאתר A לאתר B,
כהצבעה של A ל-B אבל גוגל בוחן יותר מאשר
את מספר ההצבעות שכל אתר מקבל, כלומר
מספר קישורים שאתר מסוים מקבל; בנוסף,
גוגל מנתח את האתר המצביע. הצבעות מאתר
"חשוב" הן בעלות משקל גבוה יותר, ועוזרות
לקדם אתרים אחרים."

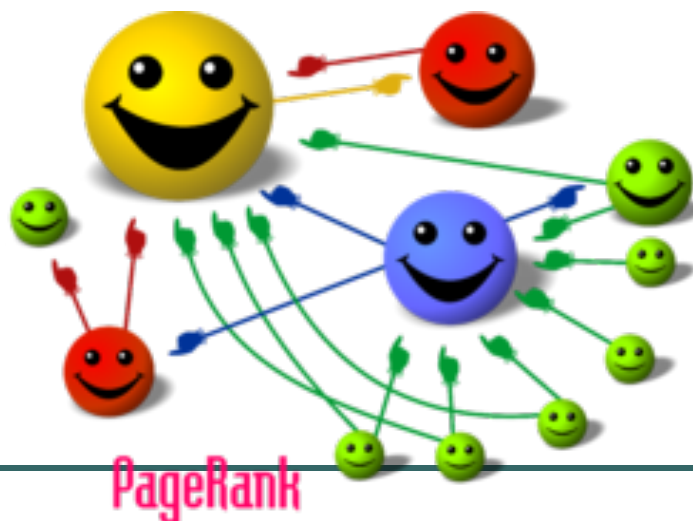
A graph model of Internet





ייצוג גרפי של העקרונות של PageRank

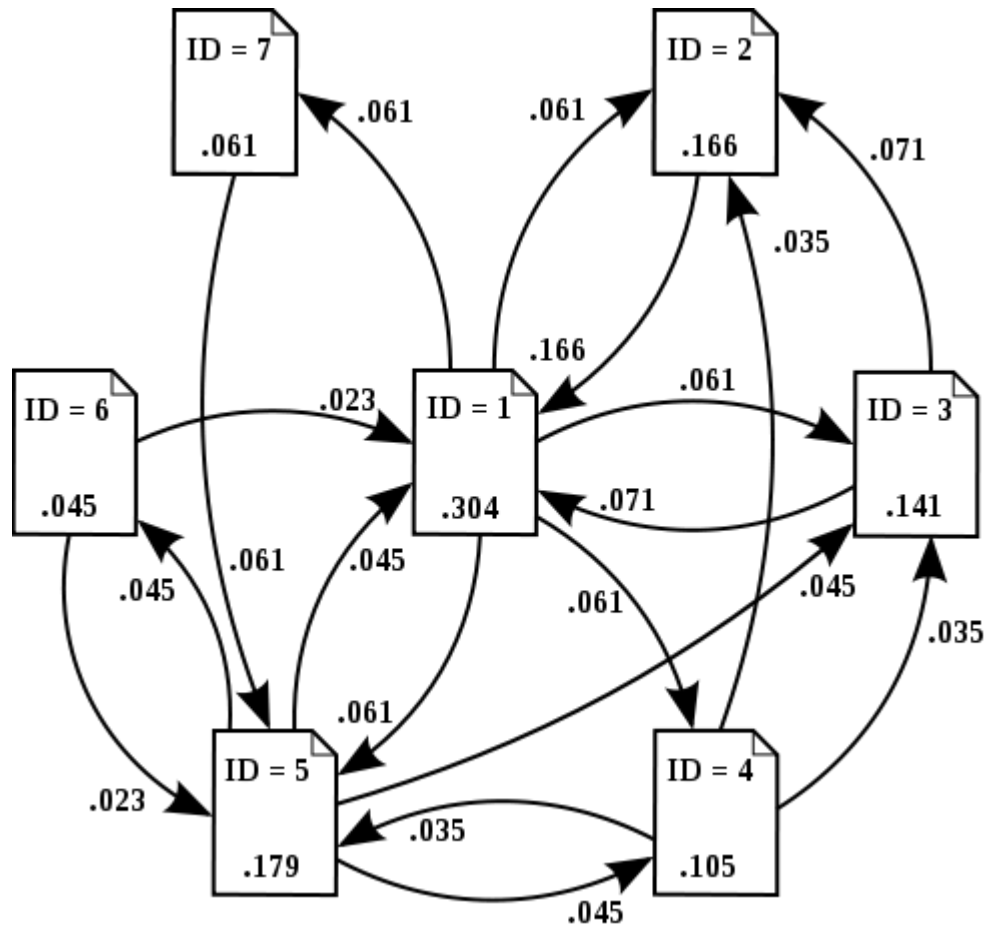
- The figure illustrates the basic principle of PageRank: The size of each node is proportional to the total size of the other faces which are pointing to it and to the amount of those links.



The Main Idea of PageRank

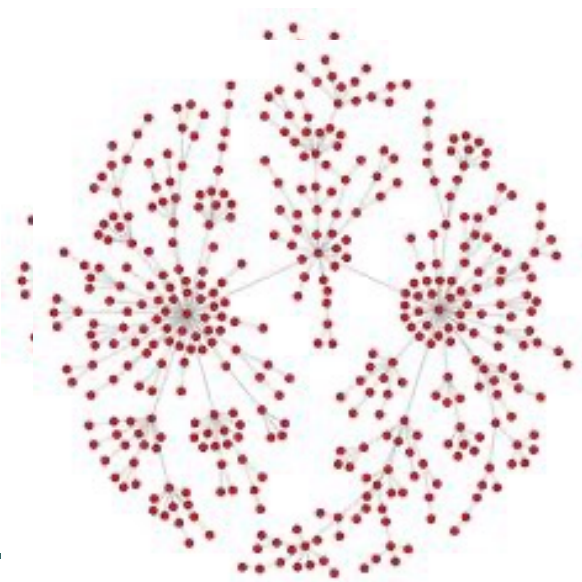
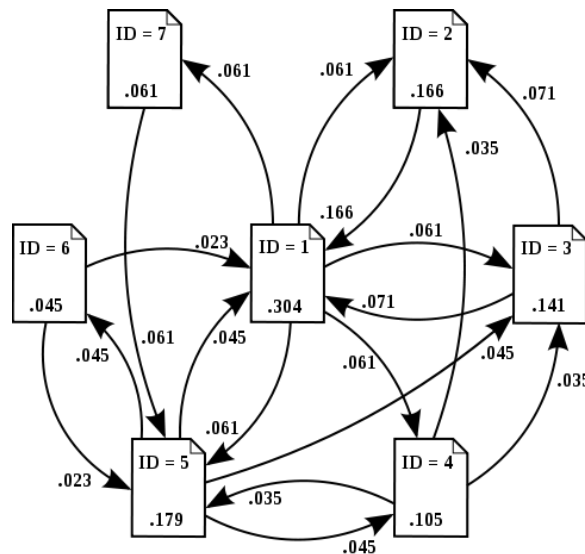
- A link to a page counts as a vote of support. The PageRank of a page is **defined recursively** and depends on the number and PageRank weights of all pages that **link to it** ("incoming links"). A page that is linked to it by **many** pages with high PageRank receives a **high rank** itself.

תרשים המציג את אופן חישובו של **PageRank**



זחלן הרשת

זחלן הרשת (crawler, bot) סורק את כל הדפים במרחב האינטרנט ויוצר מבנה היררכי של כל הקישורים בין דפים.

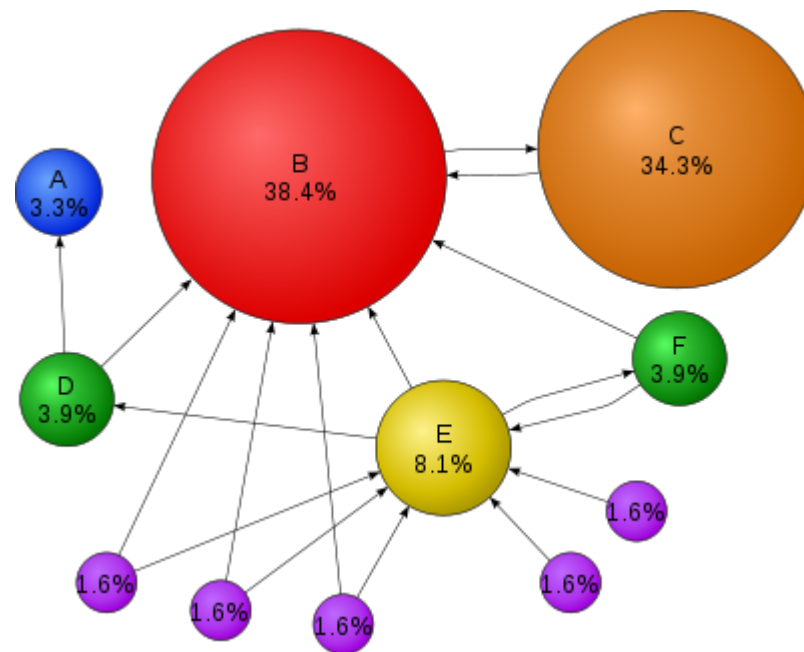


Many names for the crawler

- Crawler זחלן
 - Spider עֶפְבִּישׁ
 - Web Robot (or bot)
 - Web agent
 - Wanderer, worm, ... נוֹדֵד
-
- And famous instances: googlebot, scooter, slurp, msnbot, ...

סיכום: האלגוריתם מחשב עבור כל דף אינטרנט את מידת החשיבות שלו

- ומציג אותו בדירוג תוצאות החיפוש בהתאם. דירוג של דף נקבע על פי כמות הדפים המקשרים אליו וחשיבותם של הדפים המקשרים. כלומר, אם דפים רבים מקשרים אל דף מסוים, האלגוריתם קובע את מידת החשיבות של דפים אלה ומדרג את הדף על פי מידת חשיבותם. הדירוג מתבצע על סקאלה לוגריתמית עם ערכים בין 0 ל-10 ומושפע מגורמים נוספים כגון כמות הכניסות לדף והופעה של מילים רלוונטיות נוספות ההתקפה



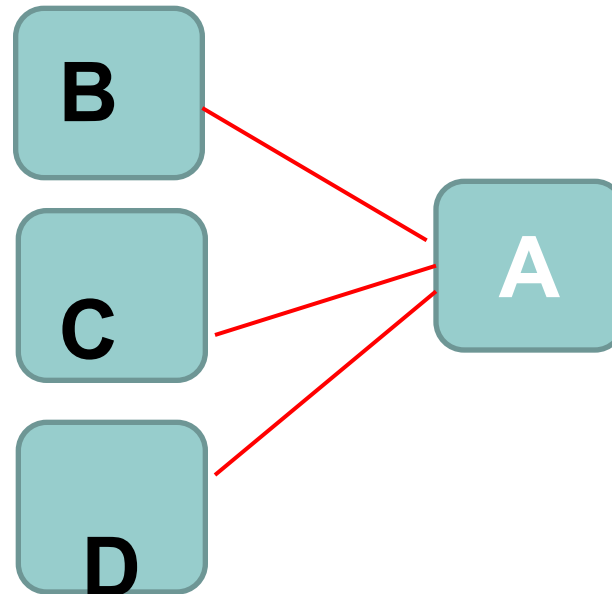
Simplified Algorithm. Examples

- Assume there are four web pages A,B,C,D. The only links in this example are from pages **B**, **C**, and **D** to **A**. **The initial value PR** for each page is taken 0.25. On the next iteration, the PageRank for a given page A is the sum of the PR's of pages corresponding to its entering to A links :

- $$PR(A)=PR(B)+PR(C)+PR(D) =0.75$$

Example 1

- B →
- C → A
- D →
-
-
-
-



General Rule #1

- $PR(A) = PR(B)/L(B) + PR(C)/L(C) + PR(D)/L(D)$
where $L(x)$ is the number of outbound links of page x .

-
- $PR(A)=$
 - $PR(B)/L(B) + PR(C)/L(C) + PR(D)/L(D)$

Rule #1. Distribution of *PR* between several nodes

- On each next iteration, the PageRank value , denoted PR , when transferred from any given page to the nodes of its outbound links is divided equally among all outbound links.
 - בכל איטרציה הבאה, ערך של PageRank, מסומן PR , כאשר מועבר מכל דף נתון לצמתים של הקישורים היוצאים שלו מחולק שווה בין כל הקישורים היוצאים

In the general case, the PageRank value for page u is:

$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)}$$

- ,i.e. the PageRank value for u is dependent on the PageRank values for each page v contained in the set B_u (the set containing all pages linking to page u), divided by the number $L(v)$ of links from page v .

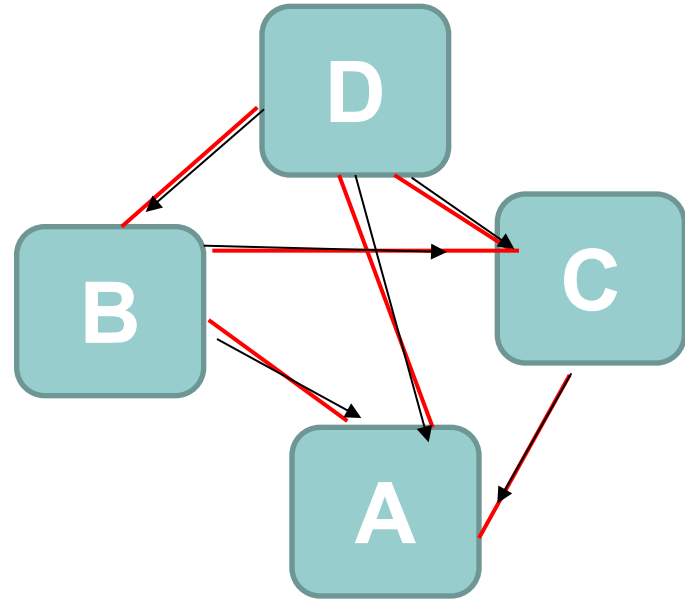
Another example .

- Suppose now that page **B** had a link to pages **C** and **A**, page **C** had a link to page **A**, and page **D** had links to all three pages. Thus, upon the first iteration, page **B** would transfer half of its existing value, or 0.125, to page **A** and the other half, or 0.125, to page **C**. Page **C** would transfer all of its existing value, 0.25, to the only page it links to, **A**. Since **D** had three outbound links, it would transfer one third of its existing value, or approximately 0.083, to **A**. At the completion of this iteration, page **A** will have a PageRank of approximately 0.458.

- $$PR(A) = PR(B)/2 + PR(C)/1 + PR(D)/3 = 0.458$$

Suppose now that page B had a link to pages C and A, page C had a link to A, and page D had links to all three pages. Thus, upon the first iteration, page B would transfer half of its existing value, or 0.125, to page A and the other half, or 0.125, to page C. Page C would transfer all of its existing value, 0.25, to the only page it links to, A. Since D had three outbound links, it would transfer one third of its existing value, or approximately 0.083, to A. At the completion of this iteration, page A will have a PageRank of approximately 0.458.

$$PR(A) = PR(B)/2 + PR(C)/1 + PR(D)/3 = 0.458$$



Rule 2 : Damping factor

גורם דעיכה

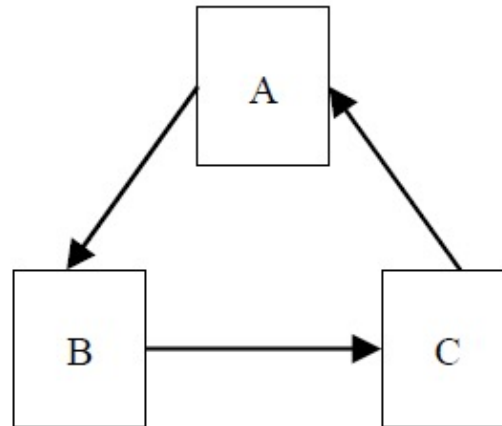
$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

where p_1, p_2, \dots, p_N are the pages under consideration, $M(p_i)$ is the set of pages that link to p_i , $L(p_j)$ is the number of outbound links on page p_j , and N is the total number of pages.

The damping factor is the probability, that at any step of the algorithm, the surfing person will continue the search

In simple words, if $d = 0,85$ then 850 sites out of 1,000 possible sites will be surfed before the search is stopped.

Example 2



The number of web pages $N = 3$

The damping parameter $d = 0.7$

$$PR(A) = (1 - d) \times (1 / N) + d \times (PR(C) / 1)$$

$$PR(B) = (1 - d) \times (1 / N) + d \times (PR(A) / 1)$$

$$PR(C) = (1 - d) \times (1 / N) + d \times (PR(B) / 1)$$

Ex. 2 (cont-d)

So

$$PR(A) = 0.1 + 0.7 \times PR(C)$$

$$PR(B) = 0.1 + 0.7 \times PR(A)$$

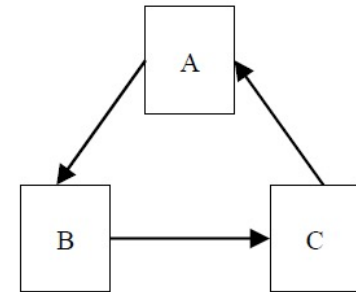
$$PR(C) = 0.1 + 0.7 \times PR(B)$$

By solving the above system of linear equations, we get

$$PR(A) = 1/3 = 0.33$$

$$PR(B) = 1/3 = 0.33$$

$$PR(C) = 1/3 = 0.33$$



The number of web pages $N = 3$

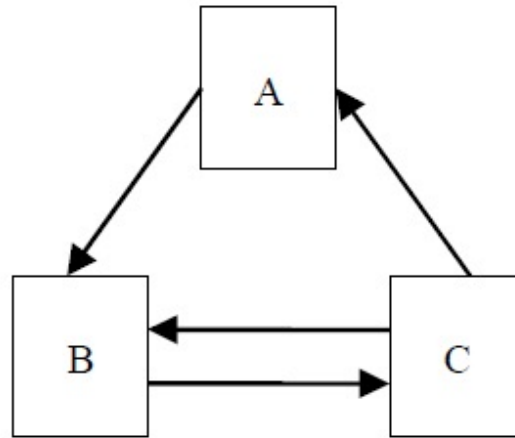
The damping parameter $d = 0.7$

$$PR(A) = (1 - d) \times (1 / N) + d \times (PR(C) / 1)$$

$$PR(B) = (1 - d) \times (1 / N) + d \times (PR(A) / 1)$$

$$PR(C) = (1 - d) \times (1 / N) + d \times (PR(B) / 1)$$

Example 3.



The number of web pages $N = 3$

The damping parameter $d = 0.7$

$$PR(A) = (1 - d) \times (1 / N) + d \times (PR(C) / 2)$$

$$PR(B) = (1 - d) \times (1 / N) + d \times (PR(A) / 1 + PR(C) / 2)$$

$$PR(C) = (1 - d) \times (1 / N) + d \times (PR(B) / 1)$$

Example 3 (cont-d)

So

$$PR(A) = 0.1 + 0.35 \times PR(C)$$

$$PR(B) = 0.1 + 0.70 \times PR(A) + 0.35 \times PR(C)$$

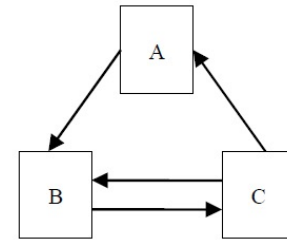
$$PR(C) = 0.1 + 0.70 \times PR(B)$$

By solving the above system of linear equations, we get

$$PR(A) = 0.2314$$

$$PR(B) = 0.3933$$

$$PR(C) = 0.3753$$



The number of web pages $N = 3$

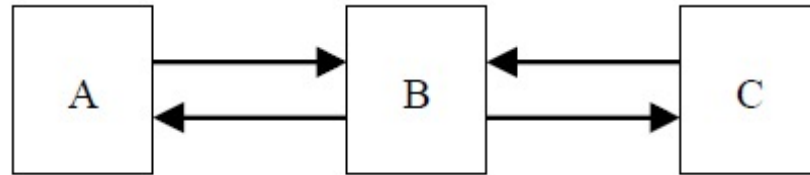
The damping parameter $d = 0.7$

$$PR(A) = (1 - d) \times (1 / N) + d \times (PR(C) / 2)$$

$$PR(B) = (1 - d) \times (1 / N) + d \times (PR(A) / 1 + PR(C) / 2)$$

$$PR(C) = (1 - d) \times (1 / N) + d \times (PR(B) / 1)$$

Example 4



The number of web pages $N = 3$

The damping parameter $d = 0.7$

$$PR(A) = (1 - d) \times (1 / N) + d \times (PR(B) / 2)$$

$$PR(B) = (1 - d) \times (1 / N) + d \times (PR(A) / 1 + PR(C) / 1)$$

$$PR(C) = (1 - d) \times (1 / N) + d \times (PR(B) / 2)$$

So

$$PR(A) = 0.1 + 0.35 \times PR(B)$$

$$PR(B) = 0.1 + 0.70 \times PR(A) + 0.70 \times PR(C)$$

$$PR(C) = 0.1 + 0.35 \times PR(B)$$

By solving the above system of linear equations, we get

Ex. 4 (cont-d)

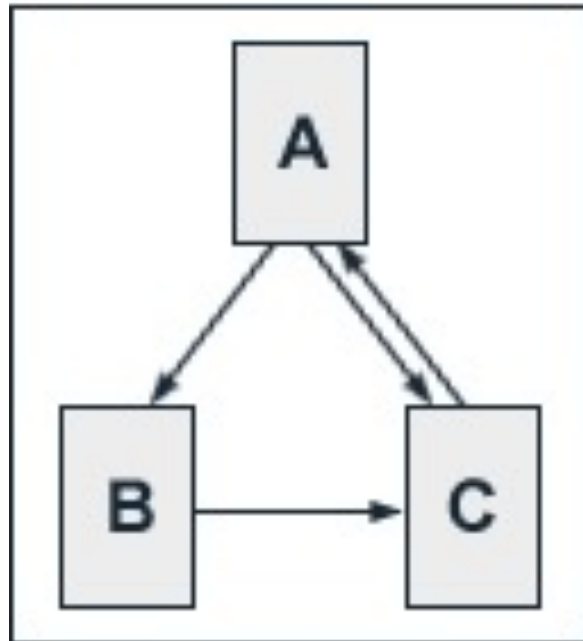
By solving the above system of linear equations, we get

$$PR(A) = 0.2647$$

$$PR(B) = 0.4706$$

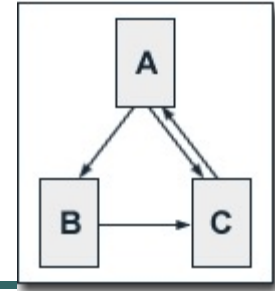
$$PR(C) = 0.2647$$

Example 5



Class work : To solve Ex. 5

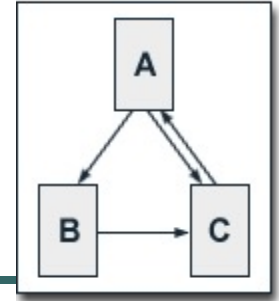
Ex.5 (Algebraic Method)



- $PR(A) = 0.5/3 + 0.5 PR(C)$
 $PR(B) = 0.5/3 + 0.5 (PR(A) / 2)$
 $PR(C) = 0.5/3 + 0.5 (PR(A) / 2 + PR(B))$
- These equations can easily be solved.
We get the following PageRank values for the single pages:
- $PR(A) = 14/13 = 1.07692308$
 $PR(B) = 10/13 = 0.76923077$
 $PR(C) = 15/13 = 1.15384615$

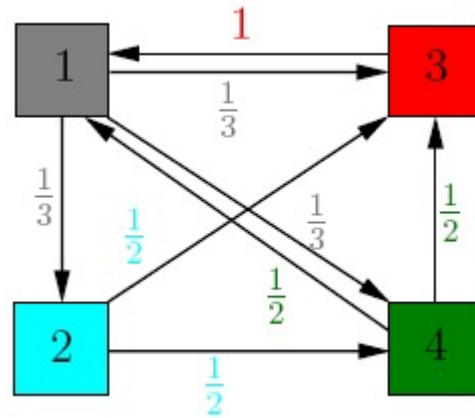
Ex 5 (method 2)

Iterative method



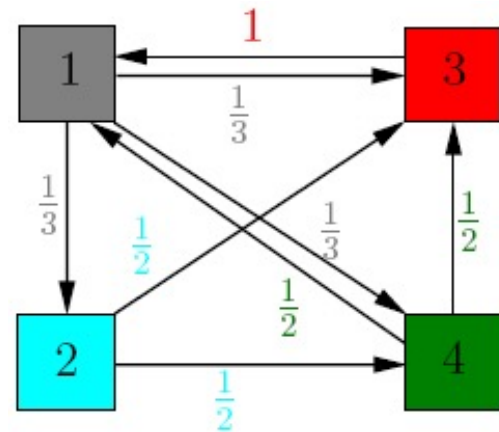
Iteration	PR(A)	PR(B)	PR(C)
0	1	1	1
1	1	0.5	1.5
2
3	1.07421875	0.76855469	1.15283203
4	1.07641602	0.76910400	1.15365601
5	1.07682800	0.76920700	1.15381050
6	1.07690525	0.76922631	1.15383947
7	1.07691973	0.76922993	1.15384490
8	1.07692245	0.76923061	1.15384592
9	1.07692296	0.76923074	1.15384611
10	1.07692305	0.76923076	1.15384615
11	1.07692307	0.76923077	1.15384615
12	1.07692308	0.76923077	1.15384615

Example 6 (matrix method 3)



- Let us denote by A the transition matrix
- $A = \{a_{ij}\} = 1/L_{ij}$, if there is link (j,i) ;
- 0 otherwise

- $$A \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$



Example 6

$$\begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

$$\mathbf{v} = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}, \quad \mathbf{A}\mathbf{v} = \begin{pmatrix} 0.37 \\ 0.08 \\ 0.33 \\ 0.20 \end{pmatrix}, \quad \mathbf{A}^2\mathbf{v} = \mathbf{A}(\mathbf{A}\mathbf{v}) = \mathbf{A} \begin{pmatrix} 0.37 \\ 0.08 \\ 0.33 \\ 0.20 \end{pmatrix} = \begin{pmatrix} 0.43 \\ 0.12 \\ 0.27 \\ 0.16 \end{pmatrix}$$

$$\mathbf{A}^3\mathbf{v} = \begin{pmatrix} 0.35 \\ 0.14 \\ 0.29 \\ 0.20 \end{pmatrix}, \quad \mathbf{A}^4\mathbf{v} = \begin{pmatrix} 0.39 \\ 0.11 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad \mathbf{A}^5\mathbf{v} = \begin{pmatrix} 0.39 \\ 0.13 \\ 0.28 \\ 0.19 \end{pmatrix}$$

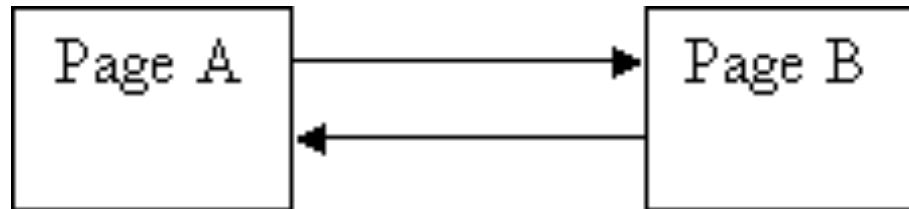
$$\mathbf{A}^6\mathbf{v} = \begin{pmatrix} 0.38 \\ 0.13 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad \mathbf{A}^7\mathbf{v} = \begin{pmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{pmatrix}, \quad \mathbf{A}^8\mathbf{v} = \begin{pmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{pmatrix}$$

Class work: To solve Example 6 with $d=0.85$

Computations. Notation revisited

- 1. **$PR(T_n)$** – importance of page (site) T_n .
- 2. **$L(T_n)$** - Each page spreads its vote out evenly amongst all of its outgoing links.
The count, or number, of outgoing links for page 1 is " $L(T_1)$ "; " $L(T_n)$ " for page n , and so on for all pages.
- 3. **$PR(T_n)/L(T_n)$** - if our page (page A) has a backlink from page " T_n ", the share of the vote page A will get is " $PR(T_n)/L(T_n)$ "

Class work: to solve the example



:

Two pages, each pointing to the other

Each page has one outgoing link
(the outgoing count is 1, i.e.

$L(A) = 1$ and $L(B) = 1$).

Guess method

$$d = 0.85$$

$$PR(A) = (1 - d) + d(PR(B)/1)$$

$$PR(B) = (1 - d) + d(PR(A)/1)$$

$$PR(A) = \frac{0.15 + 0.85 * 1}{1}$$

$$PR(B) = \frac{0.15 + 0.85 * 1}{1}$$

the numbers aren't changing at all!
So it looks like we started out with
a lucky guess!!!

A matrix form*

$$\mathbf{R} = \begin{bmatrix} PR(p_1) \\ PR(p_2) \\ \vdots \\ PR(p_N) \end{bmatrix}$$

where \mathbf{R} is the solution of the equation

$$\mathbf{R} = \begin{bmatrix} (1-d)/N \\ (1-d)/N \\ \vdots \\ (1-d)/N \end{bmatrix} + d \begin{bmatrix} \ell(p_1, p_1) & \ell(p_1, p_2) & \cdots & \ell(p_1, p_N) \\ \ell(p_2, p_1) & \ddots & & \vdots \\ \vdots & & \ell(p_i, p_j) & \\ \ell(p_N, p_1) & \cdots & & \ell(p_N, p_N) \end{bmatrix} \mathbf{R}$$

where the adjacency function $\ell(p_i, p_j)$ is 0 if page p_j does not link to p_i , and normalized such that, for each j

$$\sum_{i=1}^N \ell(p_i, p_j) = 1.$$

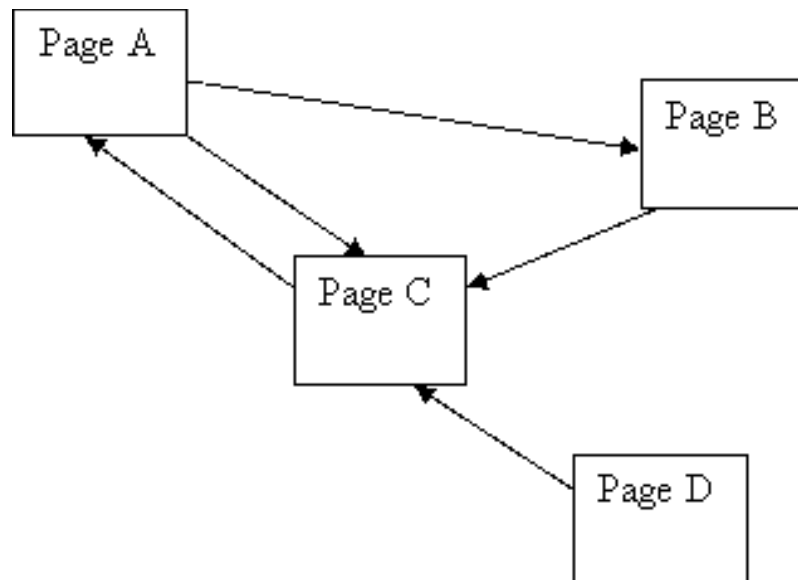
Advantages of PageRank

- Page and Brin reported that the PageRank algorithm for a network consisting of 322 million links (in-edges and out-edges) converges to within a tolerable limit in 52 iterations.
- The convergence in a network of half the above size took approximately 45 iterations.
- As a result of Markov theory, it can be shown that the PR of a page is the probability of arriving at that page after a large number of random clicks.

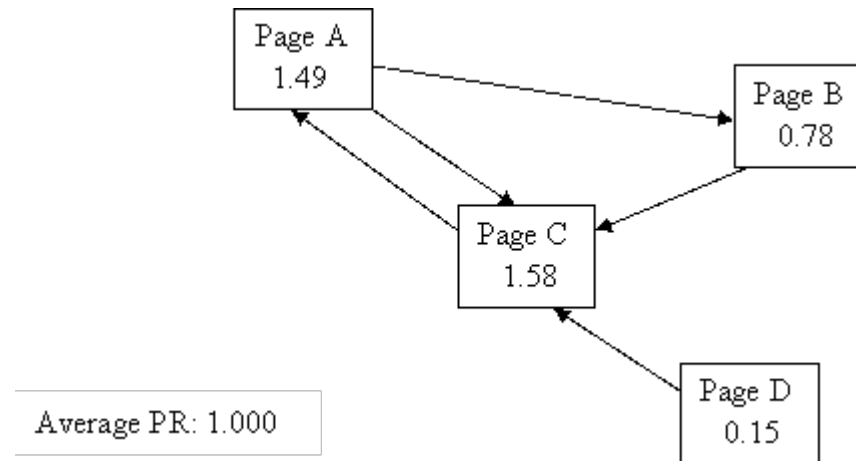
Disadvantages of PageRank

- One main disadvantage of PageRank is that it favors older pages. A new page, even a very good one, usually will not have many links.
- PageRank allows for different manipulations

Home work 1.1 Compute using three methods and $d=0.85$



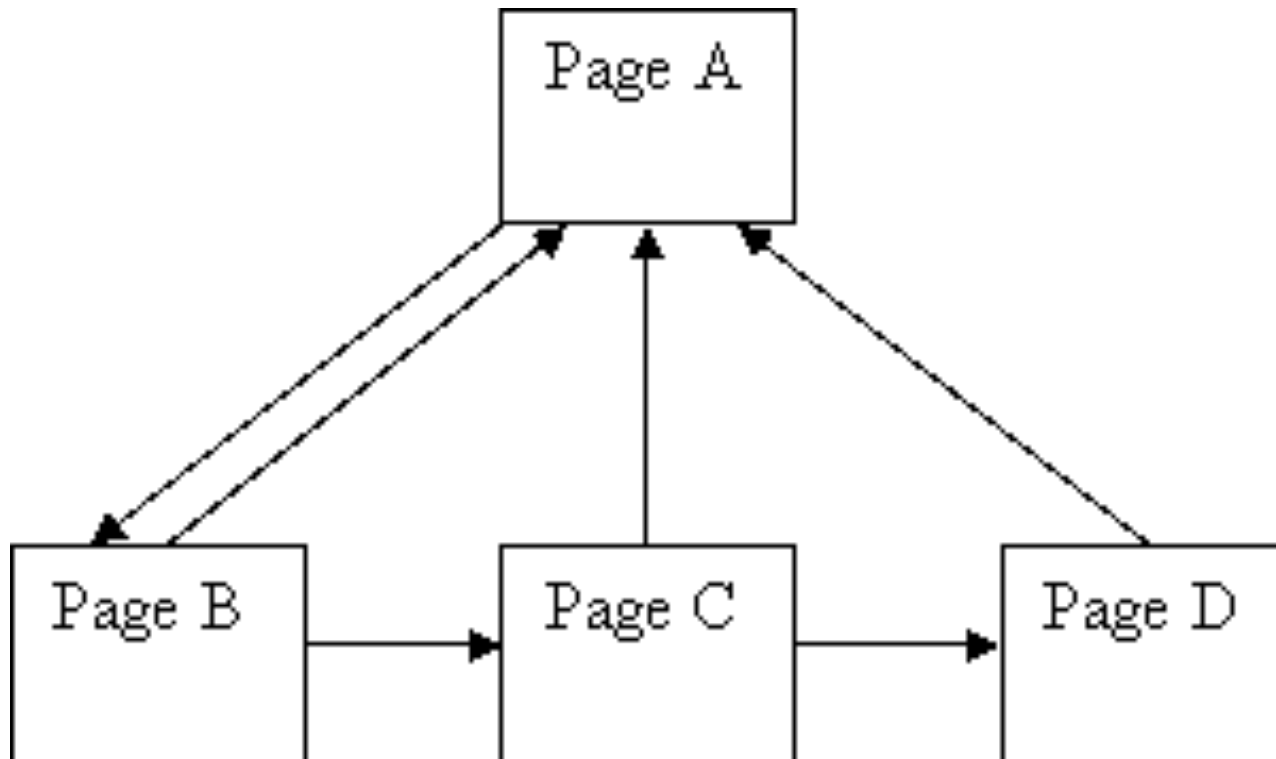
Help: The correct answer



:

You can see it took about 20 iterations

Home work 1.2. Compute using three methods and different $d=0.5; 0.7; 0.85$



- Home work 2 . Write a code for the PageRank
-

- Home work 3 . Compute PR for the graph nodes in Slide 25.

Other tasks

- 1. Advantages and disadvantages PRM
- 2. Advantages and disadvantages A^*
- 3. To solve the tasks by A^* and by Dijkstra
- 4. To solve Ex6 (l. 5), 1,2 (l.5) and in slide 25 (l.5) by 3 versions of Pagerank ($d=0,85$)

●

נראה בקרוב!