PageRank

PageRank

How does Google decide which pages to return?

Produce two rankings for each page

- Relevance ranking
- Importance ranking

Use a weighted sum of these

PageRank

Relevance Ranking

- Based on content of page
  - Words, HTML markup, etc

Importance

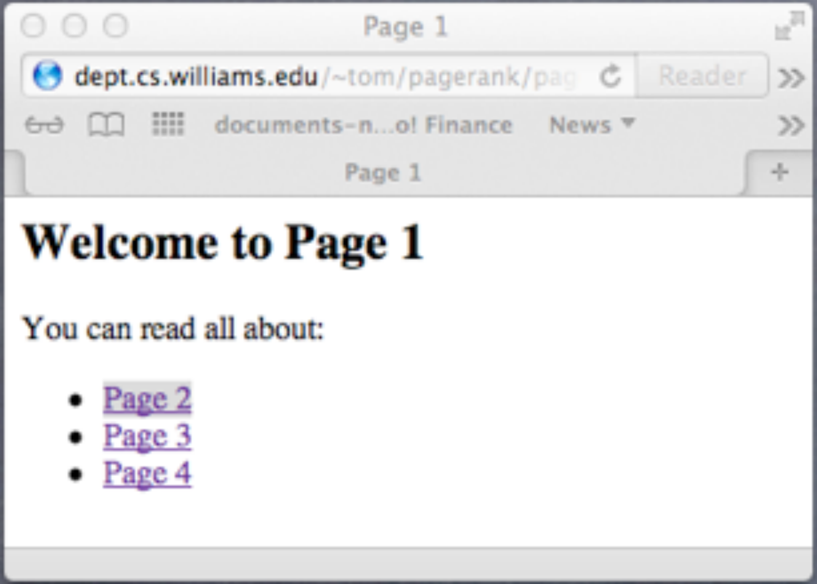
- Based on structure of the web graph

We’ll discuss the Importance Ranking

PageRank

Compute a measure R(v) for every web page v

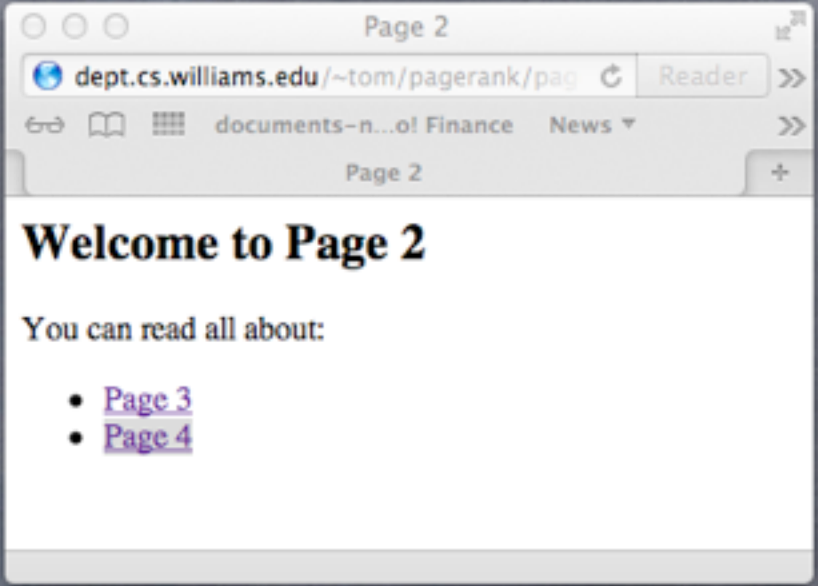
- R(v) should reflect importance of pages that link to v



Welcome to Page 1

You can read all about:

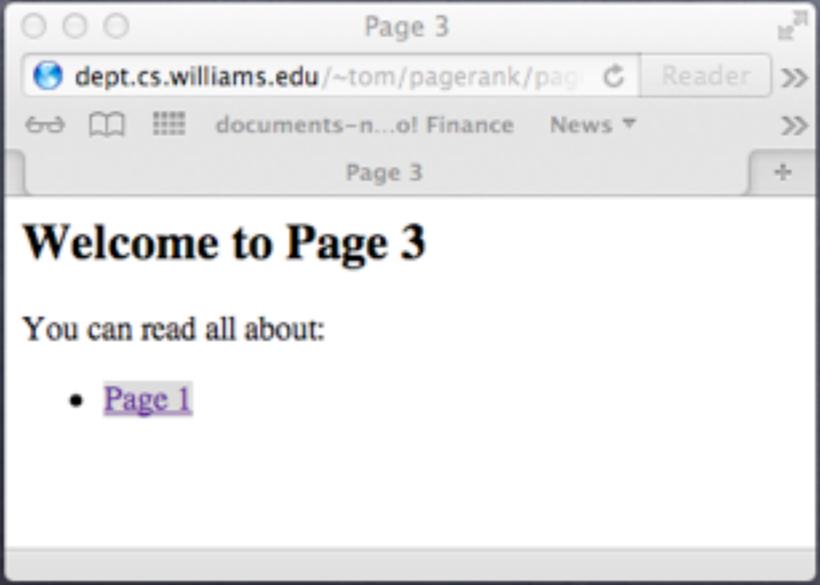
- Page 2
- Page 3
- Page 4



Welcome to Page 2

You can read all about:

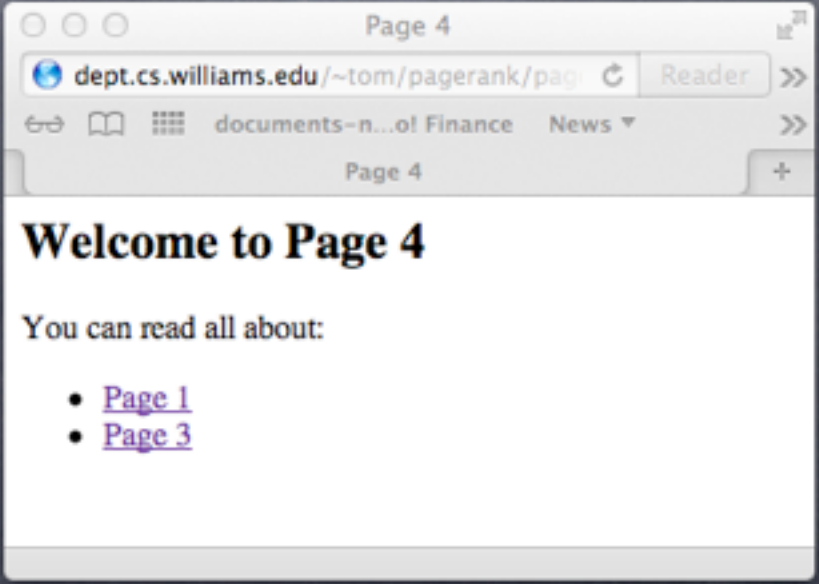
- Page 3
- Page 4



Welcome to Page 3

You can read all about:

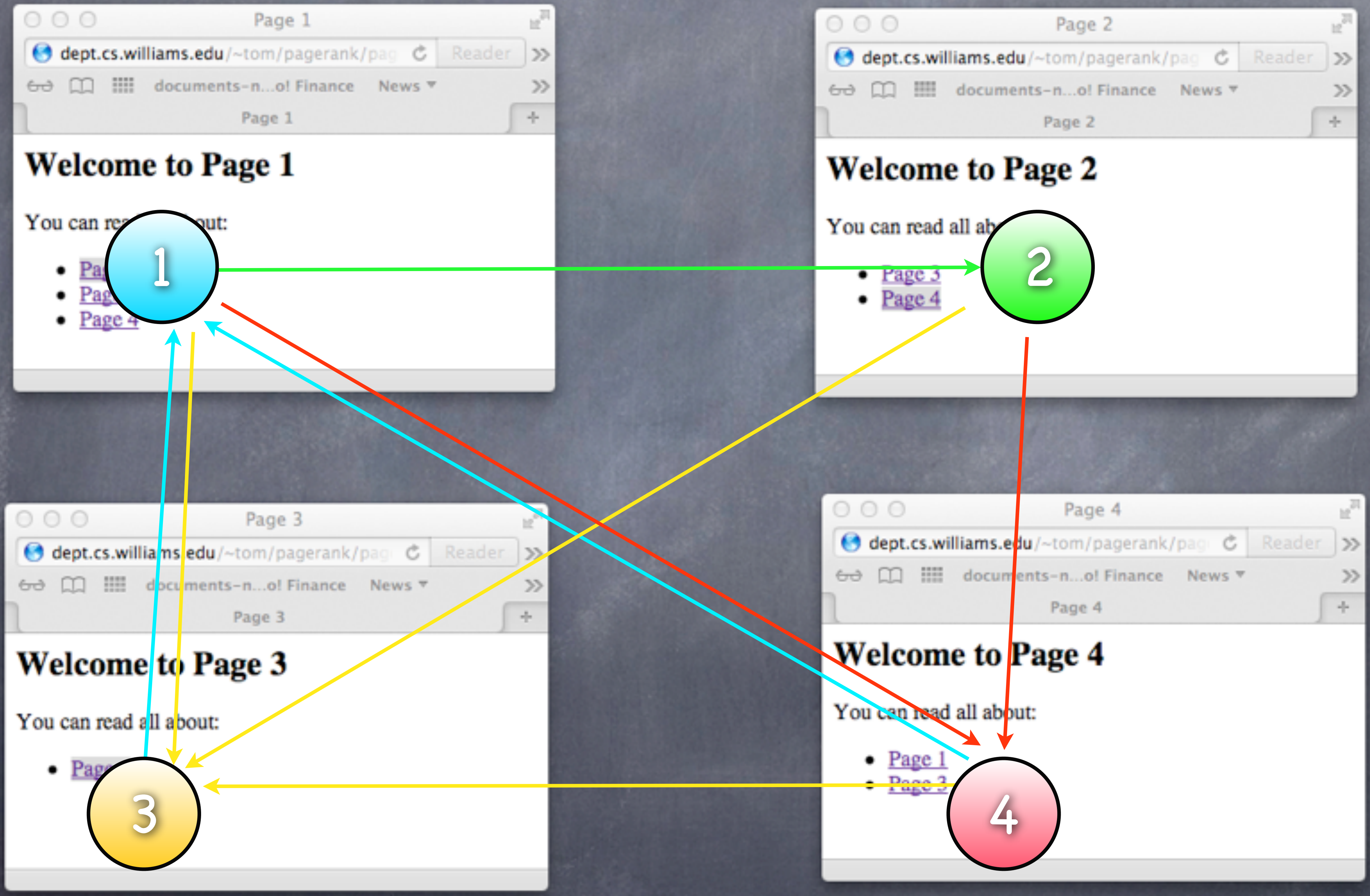
- Page 1



Welcome to Page 4

You can read all about:

- Page 1
- Page 3



Welcome to Page 1

You can re out:

- Pa
- Pag
- Page 4

Welcome to Page 2

You can read all ab

- Page 3
- Page 4

Welcome to Page 3

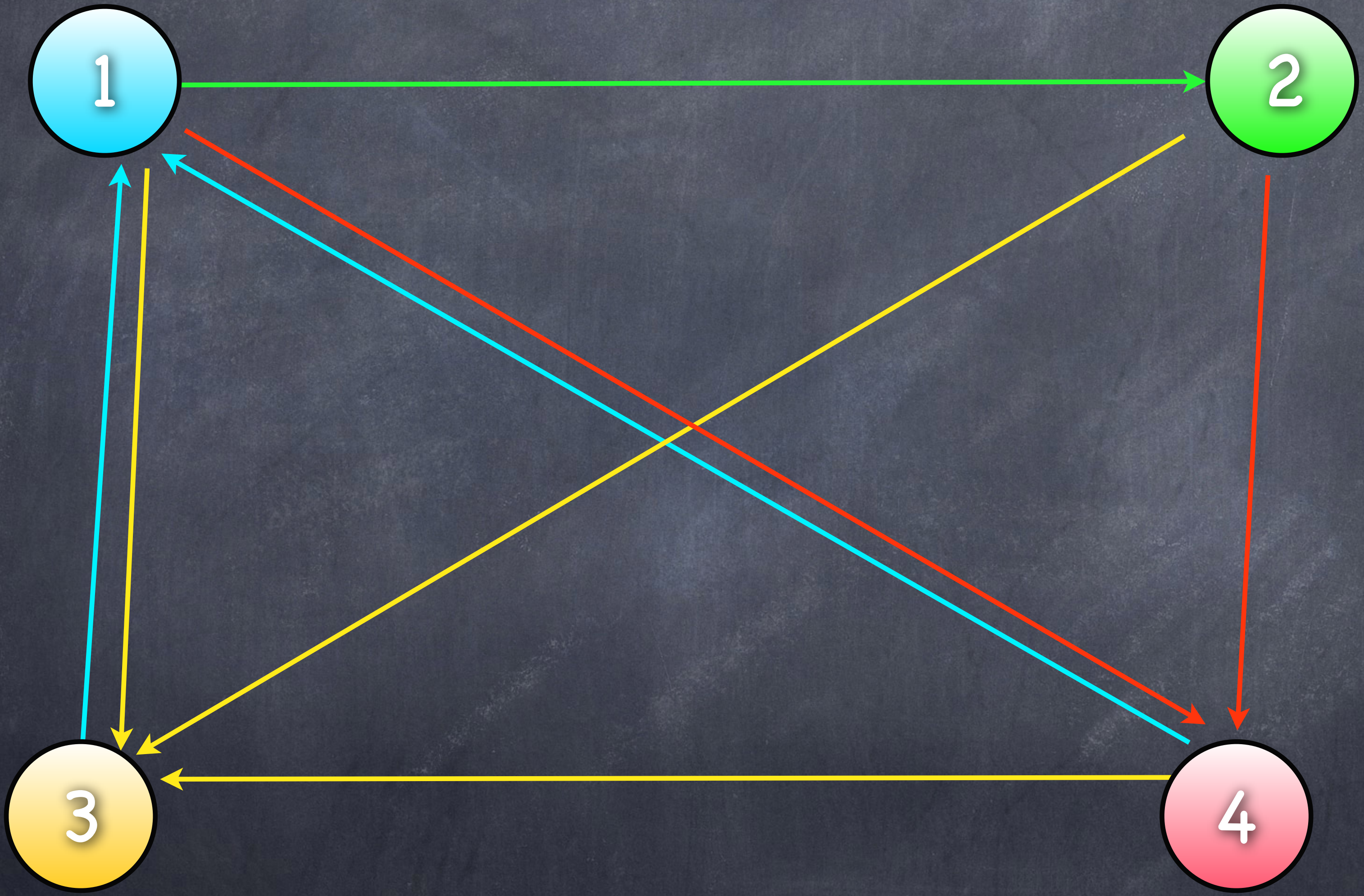
You can read all about:

- Pag

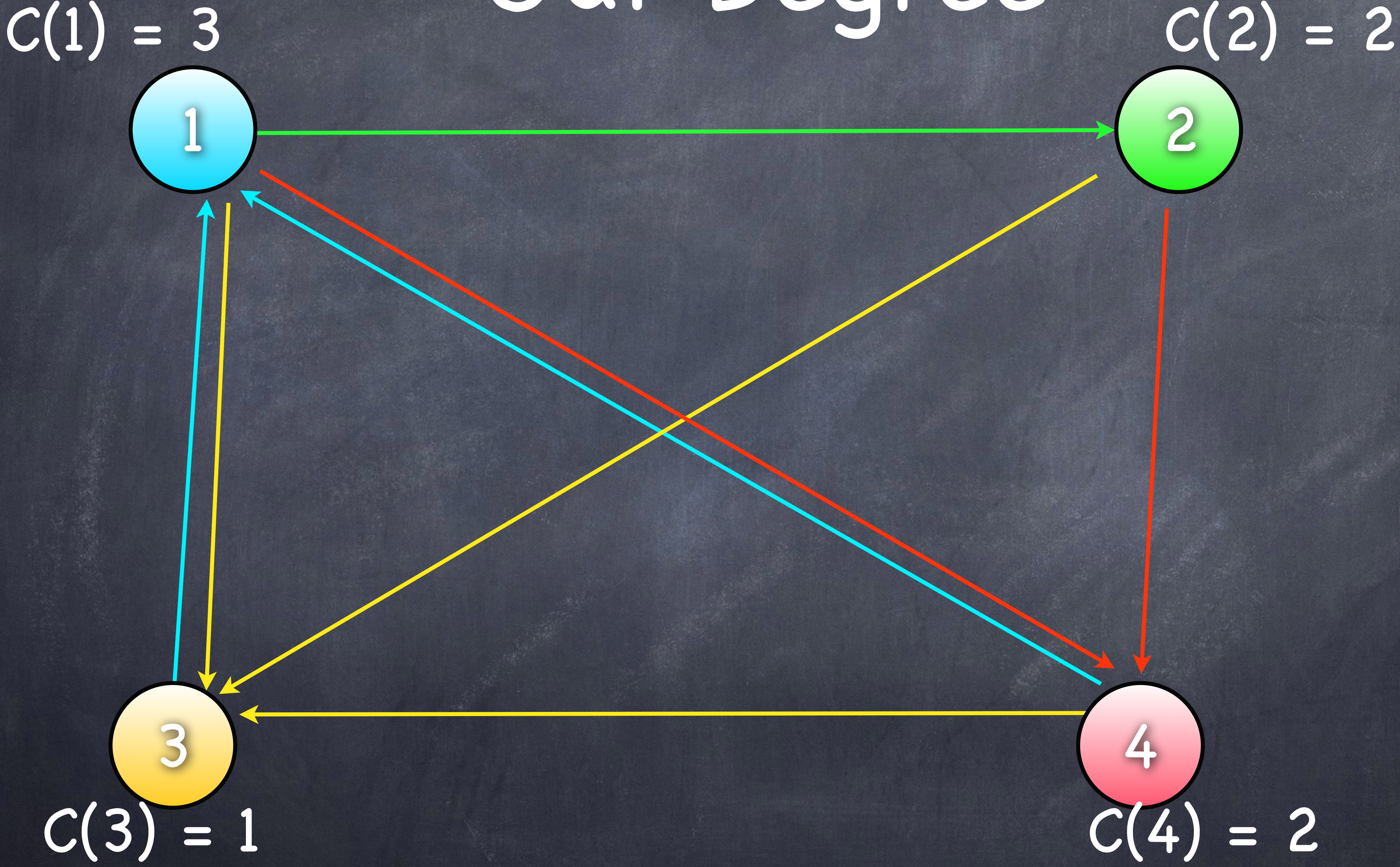
Welcome to Page 4

You can read all about:

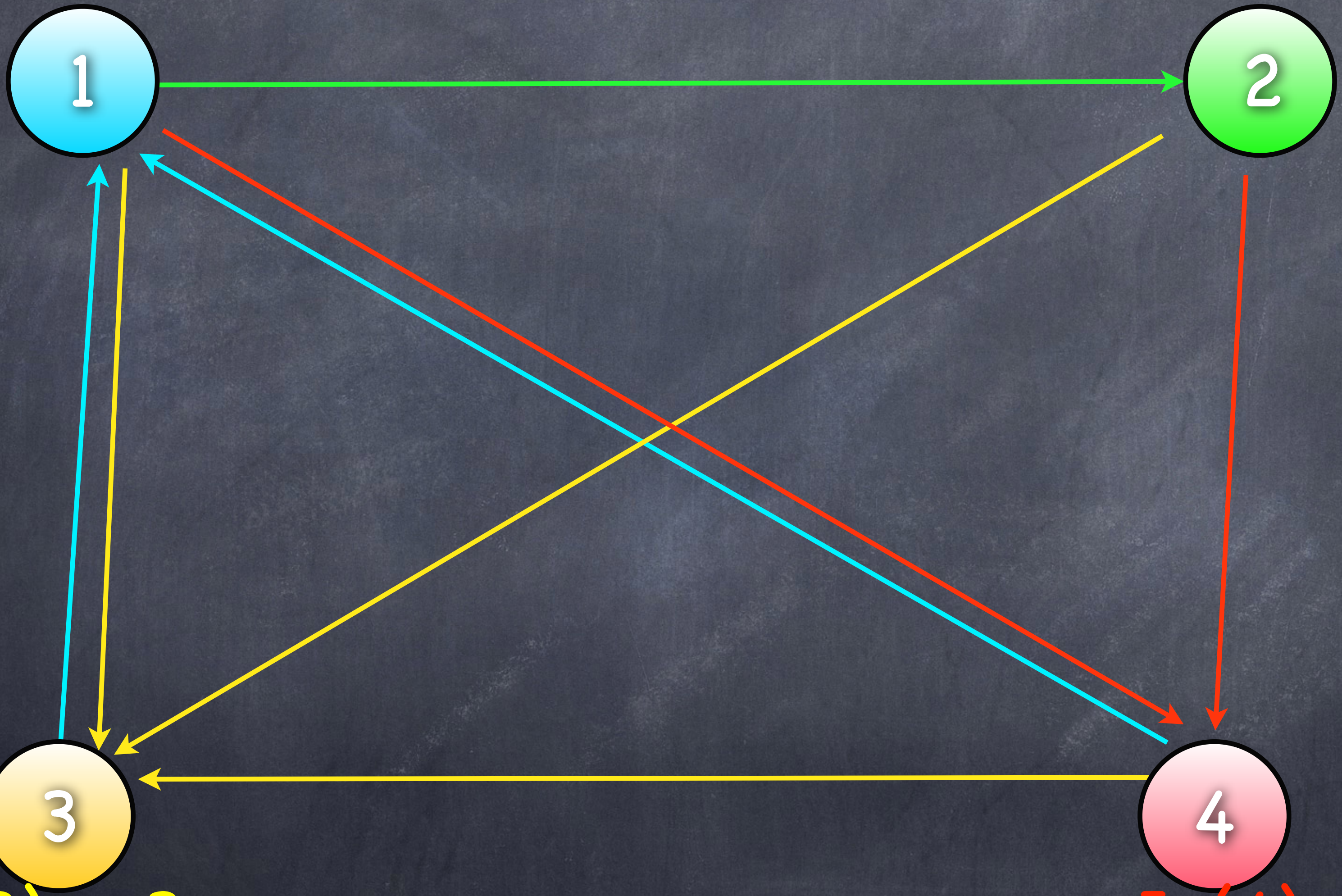
- Page 1
- Page 3


1
2
3
4

Out Degree



In Degree



In(1) = 2

In(2) = 1

In(3) = 3

In(4) = 2

PageRank : Parameters

- Parameters of web graph G
  - N & L : Number of vertices & edges in G
  - C(v) : out-degree of v (number of links **from** v)
  - R(v) : the rank of v (to be computed)
- Big idea
  - Google Juice = liquid rank

PageRank : Google Juice

- Ranking as (fluid) flow in a network
- Each page shares its importance with pages it links to
  - Page u gives each neighbor R(u)/C(u) of its importance
- So Each page gets importance from pages that link to it
  - If $u_1, \ldots, u_{In(v)}$ are pages linking to page v
  - then $R(v) = R(u_1)/C(u_1) + \ldots + R(u_{In(v)}))/C(u_{In(v)})$

PageRank : Iterated Rankings

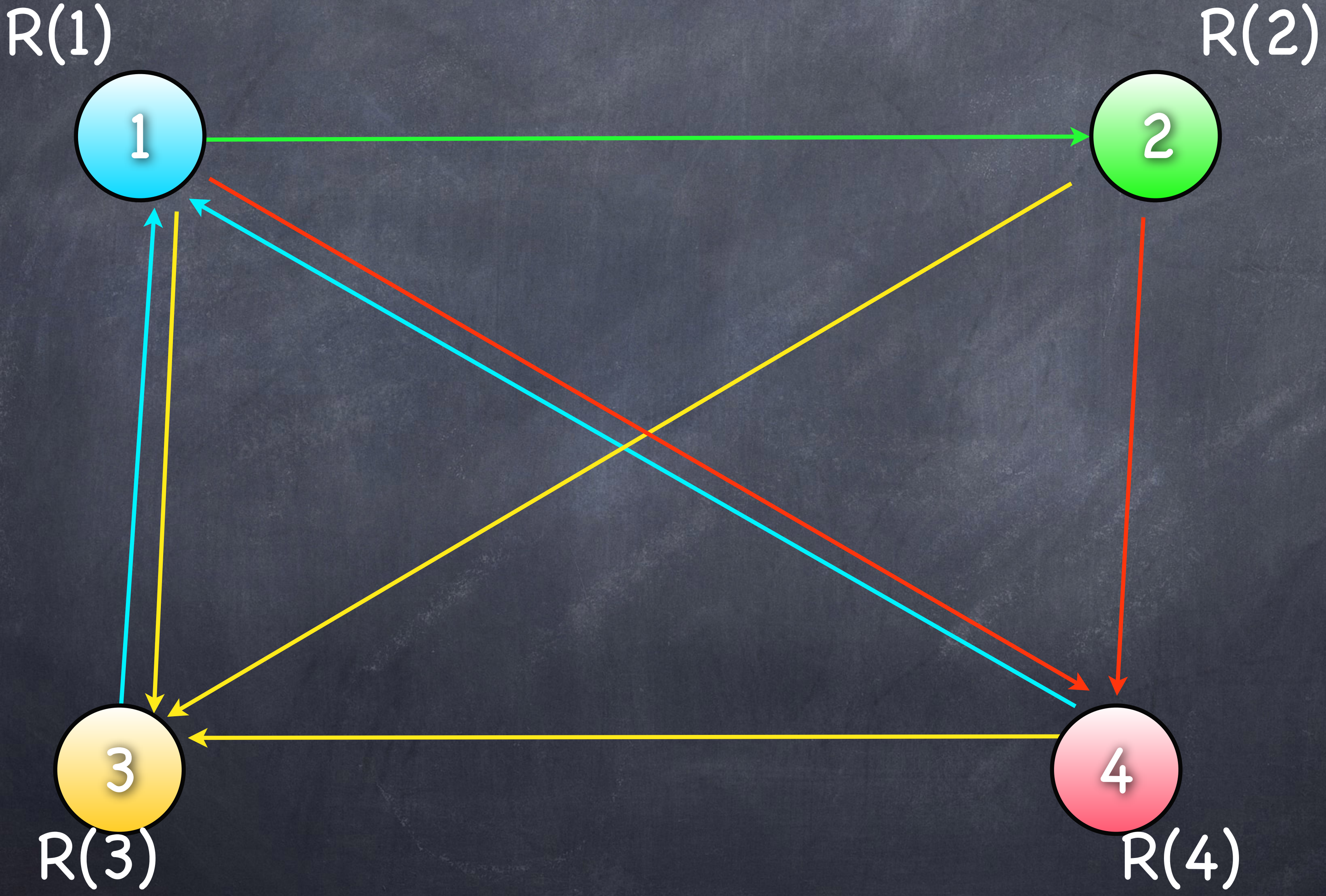
Goal: Find a ranking satisfying

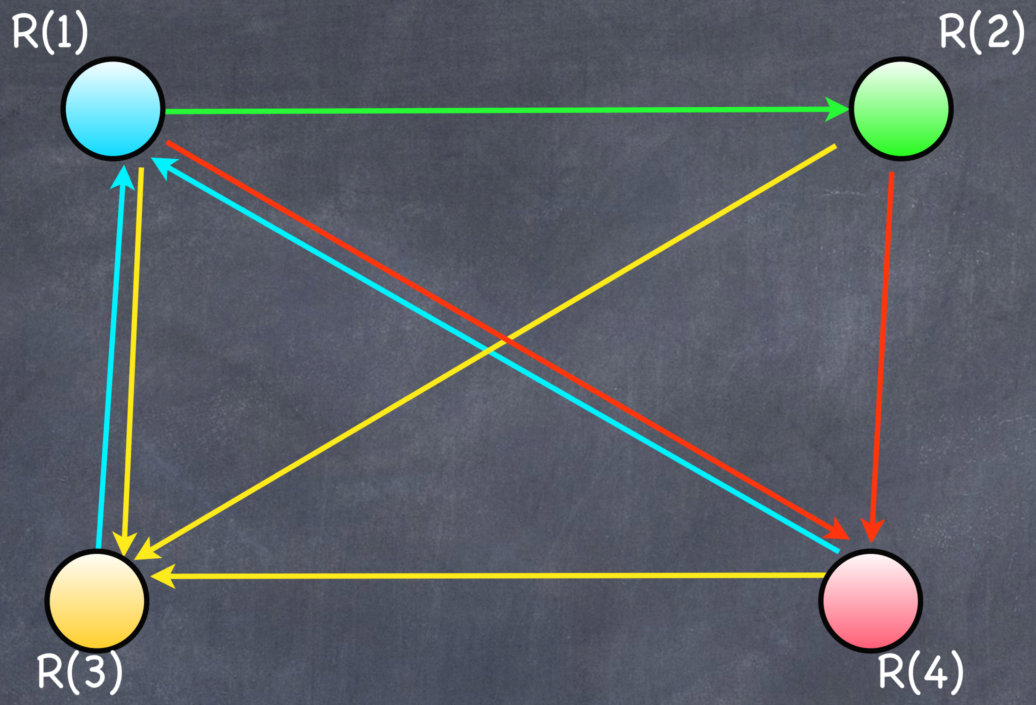
$$R(v) = R(u_1)/C(u_1) + \ldots + R(u_{In(v)}))/C(u_{In(v)})$$

The Algorithm:

- Find an initial ranking: For example, $R_0(v) = In(v)/L$
- Let Google Juice flow to give new ranking
  - $R_1(v) = R_0(u_1)/C(u_1) + \ldots + R_0(u_{In(v)}))/C(u_{In(v)})$
- Repeat many times to get rankings R_2, R_3, R_4, …
- Stop when R_n is not much different from R_{n-1}

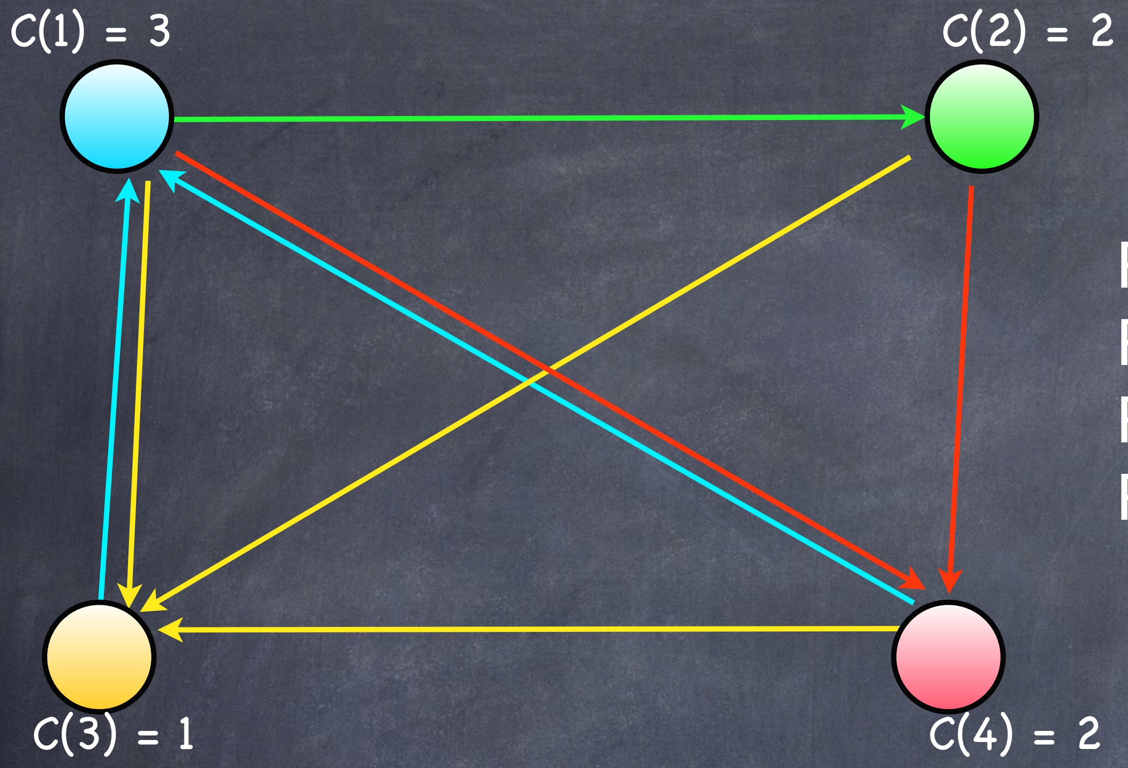
Ranking Function





$$R_1(1) = R_0(3)/C(3) + R_0(4)/C(4)$$
$$R_1(2) = R_0(1)/C(1)$$
$$R_1(3) = R_0(1)/C(1) + R_0(2)/C(2) + R_0(4)/C(4)$$
$$R_1(4) = R_0(1)/C(1) + R_0(2)/C(2)$$

$$R_1(1) = R_0(3) + R_0(4)/2$$
$$R_1(2) = R_0(1)/3$$
$$R_1(3) = R_0(1)/3 + R_0(2)/2 + R_0(4)/2$$
$$R_1(4) = R_0(1)/3 + R_0(2)/2$$



$R_1(1) = R_0(3) + R_0(4)/2$
$R_1(2) = R_0(1)/3$
$R_1(3) = R_0(1)/3 + R_0(2)/2 + R_0(4)/2$
$R_1(4) = R_0(1)/3 + R_0(2)/2$

	1	2	3	4
R_0	0.25	0.125	0.375	0.25

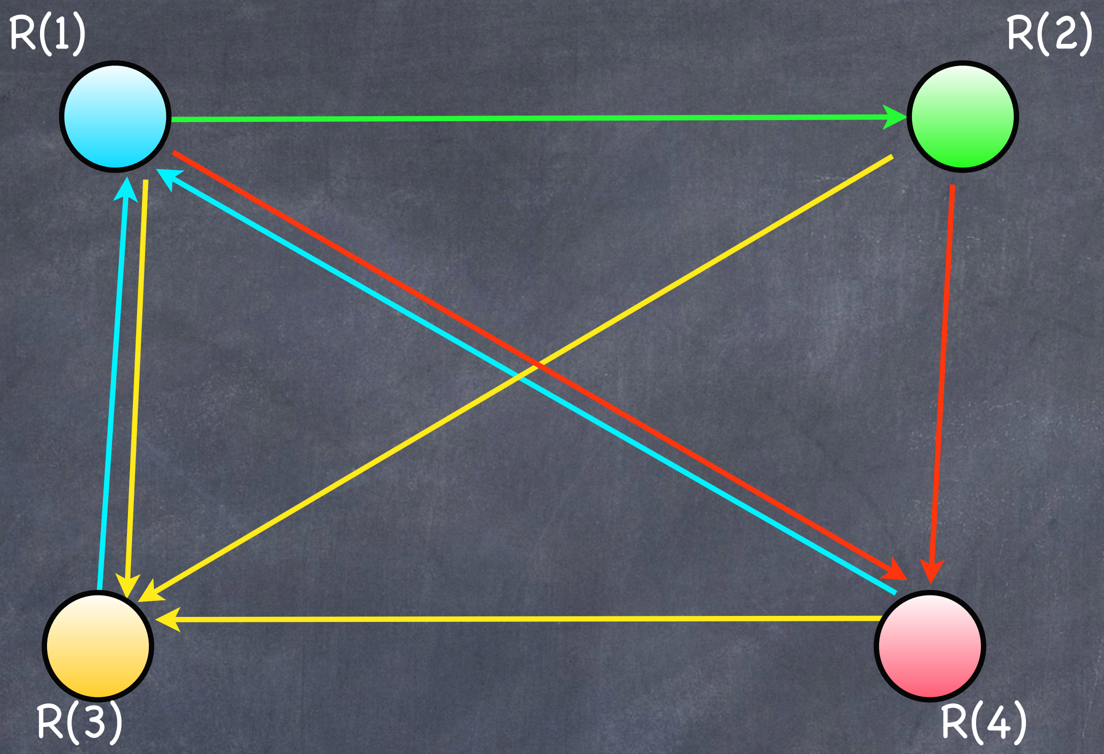
- $R_1(1) = R_0(3) + R_0(4)/2 = 0.375 + .125 = 0.5$
- $R_1(2) = R_0(1)/3 = 0.08333$
- $R_1(3) = R_0(1)/3 + R_0(2)/2 + R_0(4)/2 = 0.08333 + 0.0625 + 0.125 = 0.2708$
- $R_1(4) = R_0(1)/3 + R_0(2)/2 = 0.08333 + 0.0625 = 0.14583$

Computing Rank Functions R_n()

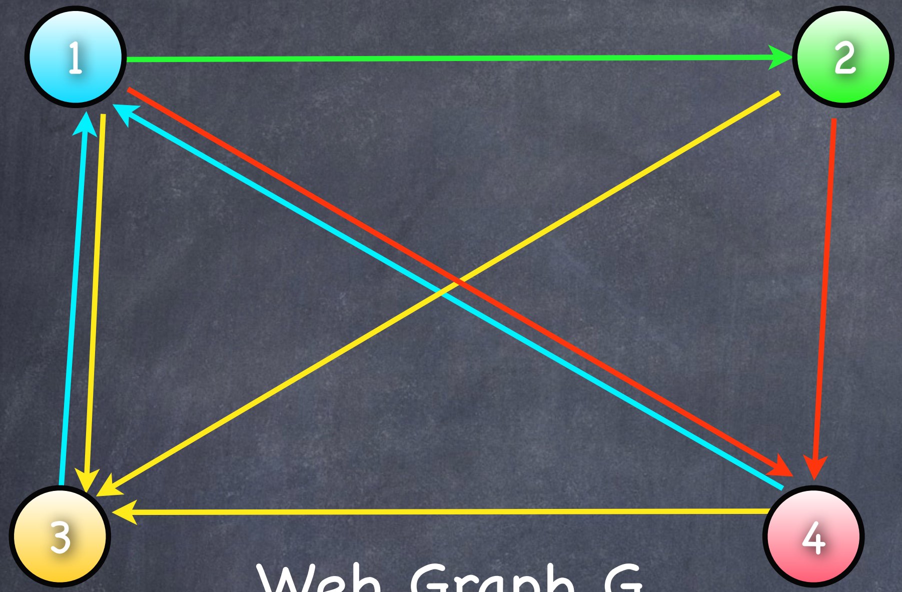
	1	2	3	4
R_0	0.25	0.125	0.375	0.25
R_1	0.5	0.083333	0.270833	0.145833
R_2	0.34375	0.166667	0.28125	0.208333
R_3	0.385417	0.114583	0.302083	0.197917
...				
R_{23}	0.387097	0.129032	0.290323	0.193548
R_{24}	0.387097	0.129032	0.290323	0.193548

PageRank : Amazing Result

- On any reasonably structured graph, this method will converge!
- Reasonably structured
  - For every pair of vertices {u,v} there is a directed path from u to v and one from v to u. [G is **strongly connected]**
  - Not all cycle-lengths are multiples of a common value k > 1 [G is **aperiodic**]



$$R(1) = R(3) + R(4)/2$$
$$R(2) = R(1)/3$$
$$R(3) = R(1)/3 + R(2)/2 + R(4)/2$$
$$R(4) = R(1)/3 + R(2)/2$$



Web Graph G

	1	2	3	4
R_0	1/4	1/8	3/8	1/4

Initial Ranking

u

v

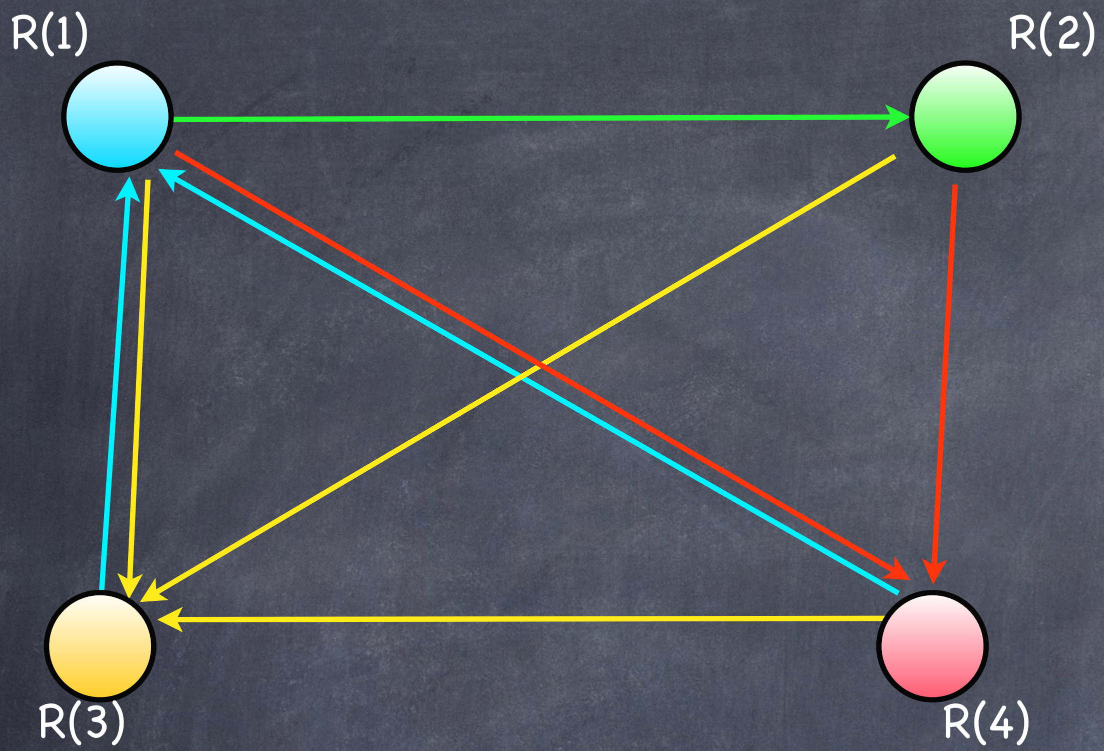
A	1	2	3	4
1	0	0	1	1/2
2	1/3	0	0	0
3	1/3	1/2	0	1/2
4	1/3	1/2	0	0

$A_{v,u} = 1/C(u)$ if $u \rightarrow v$

$A_{v,u} = 0$ otherwise

$$R_1(3) = R_0(1)/3 + R_0(2)/2 + R_0(4)/2$$

$$R_1(3) = R_0(1) * A_{3,1} + R_0(2) * A_{3,2} + R_0(3) * A_{3,3} + R_0(4) * A_{3,4}$$



$$R(1) = R(3) + R(4)/2$$
$$R(2) = R(1)/3$$
$$R(3) = R(1)/3 + R(2)/2 + R(4)/2$$
$$R(4) = R(1)/3 + R(2)/2$$

$$\begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ R(4) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{pmatrix} \begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ R(4) \end{pmatrix}$$

$$X \quad = \quad A \quad * \quad X$$

$$\begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ R(4) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{pmatrix} \begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ R(4) \end{pmatrix}$$

X = A * X

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0-1 & 0 & 1 & 1/2 \\ 1/3 & 0-1 & 0 & 0 \\ 1/3 & 1/2 & 0-1 & 1/2 \\ 1/3 & 1/2 & 0 & 0-1 \end{pmatrix} \begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ R(4) \end{pmatrix}$$

0 = (A - I) * X

PageRank as Linear Algebra

Rewrite equations

$$R(v) = R(u_1)/C(u_1) + \ldots + R(u_{In(v)}))/C(u_{In(v)})$$

Vertices : $v_1, \ldots, v_n$

Let $x_i = R(v_i)$ and let $X = (x_1, \ldots, x_n)$ then

$$x_i = x_1 \cdot A[i,1] + x_2 \cdot A[i,2] + \cdot\cdot\cdot + x_n \cdot A[i,n]$$

So $X = A \cdot X$, a matrix equation for $n \times n$ matrix A

A solution exists when A is **invertible**

PageRank as Random Walk

- Think of R_0 as a probability distribution
  - $R_0(v)$: probability of starting at v (or)
  - $R_0(v)$: probability of being at v after 0 steps

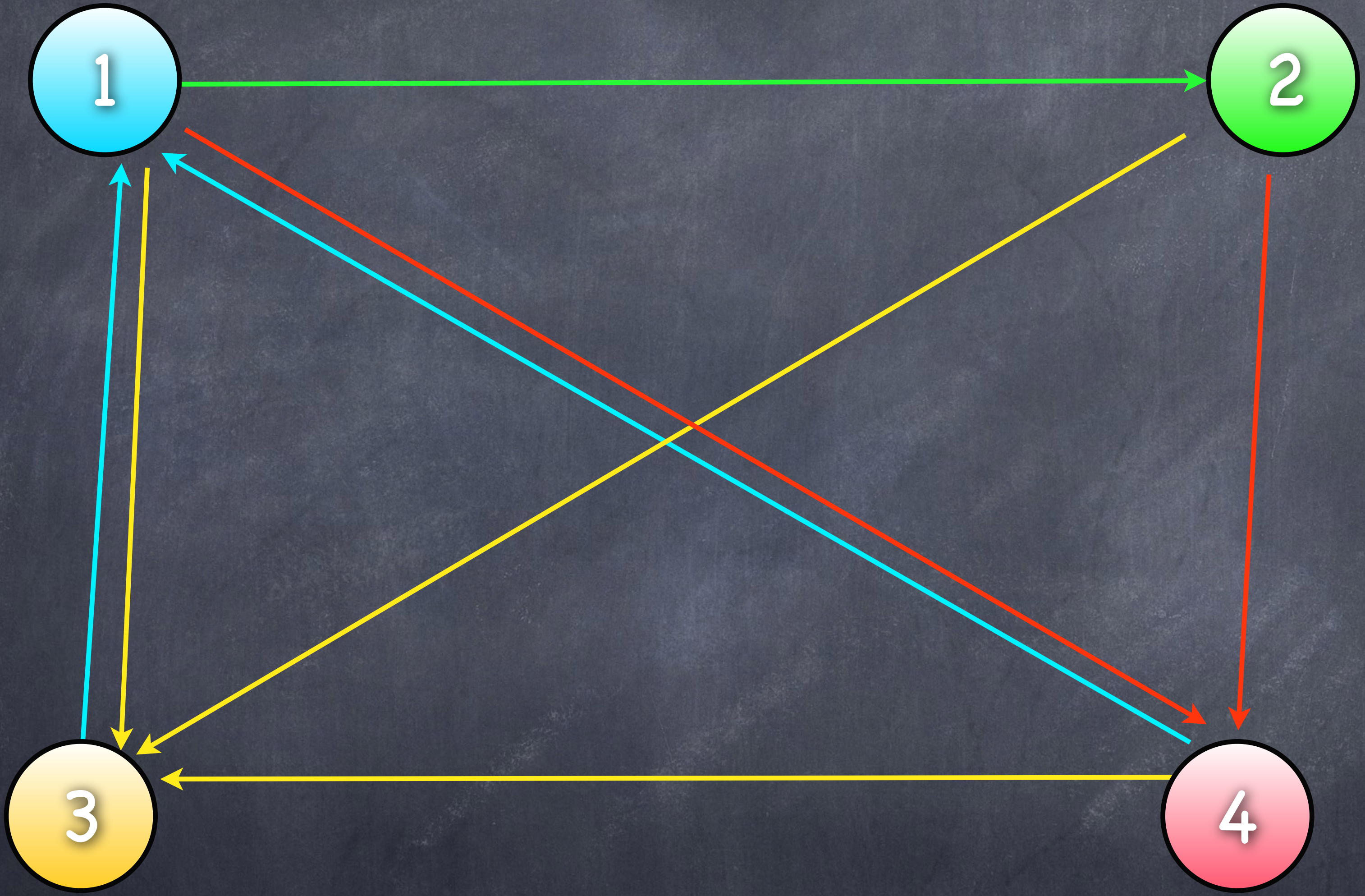
Random Walks on Graphs

- How can we interpret R_1?
- $R_1(3) = R_0(1)*A_{3,1} + R_0(2)*A_{3,2} + R_0(3)*A_{3,3} + R_0(4)*A_{3,4}$
  - $R_0(j)*A_{i,j} = R_0(j)*(1/C(j))$ (or 0)
    - Probability we were at j and then moved to i
    - Assumes equal likelihood of taking any outgoing edge
- So $R_1(3)$ is the probability that we got to vertex 3 in 1 step!
- That is: $R_1(i)$ = probability of being at page i after 1 click
  - Assuming that the starting distribution was R_0

Random Walks on Graphs

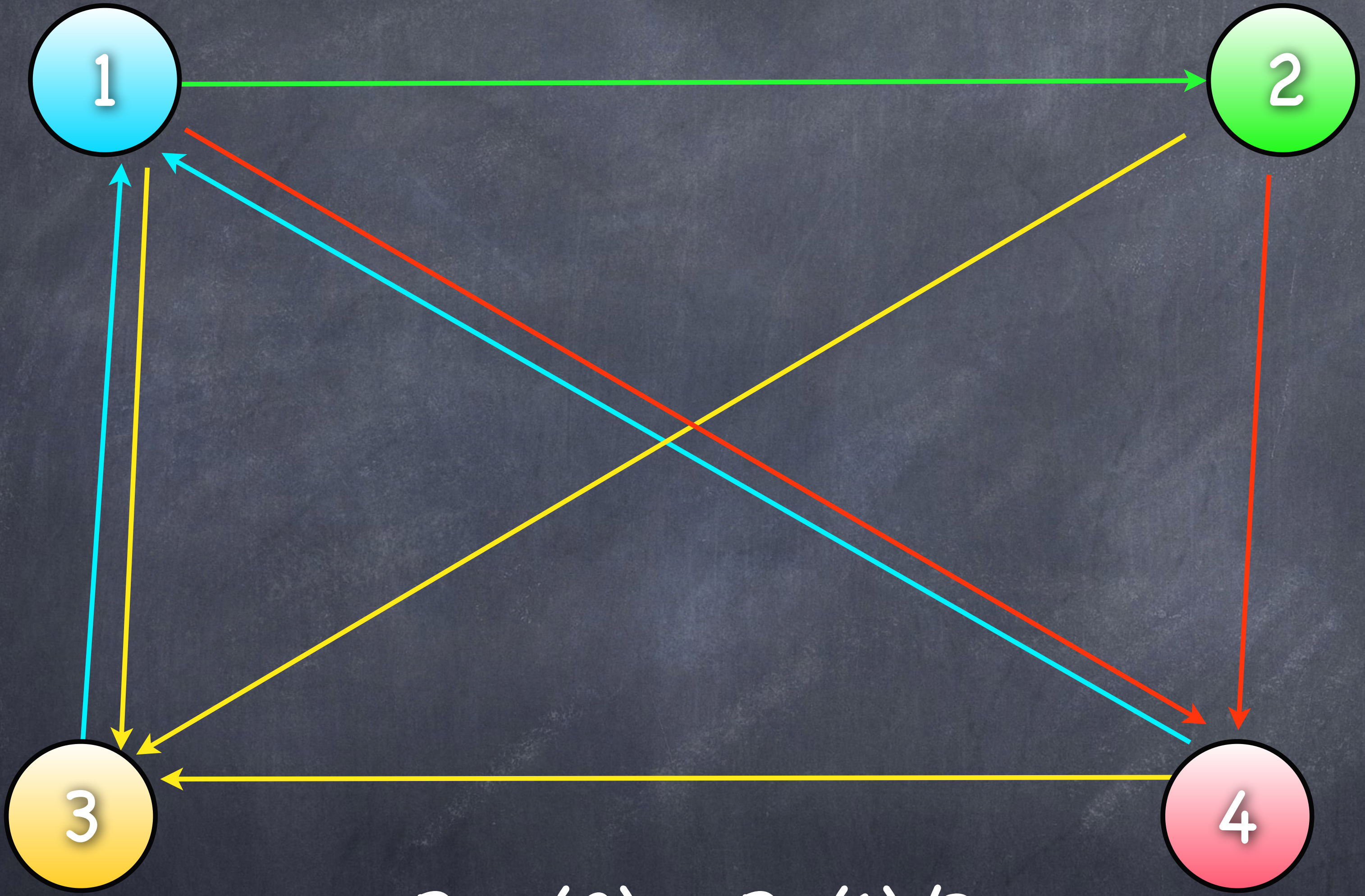
- Similarly, $R_i(j)$ is the probability of being at page j after exactly i clicks (given starting distribution R_0)
- Rename $R_i()$ to be $Pr_i()$ to emphasize this fact
- Let's try an example!

Random Surfer



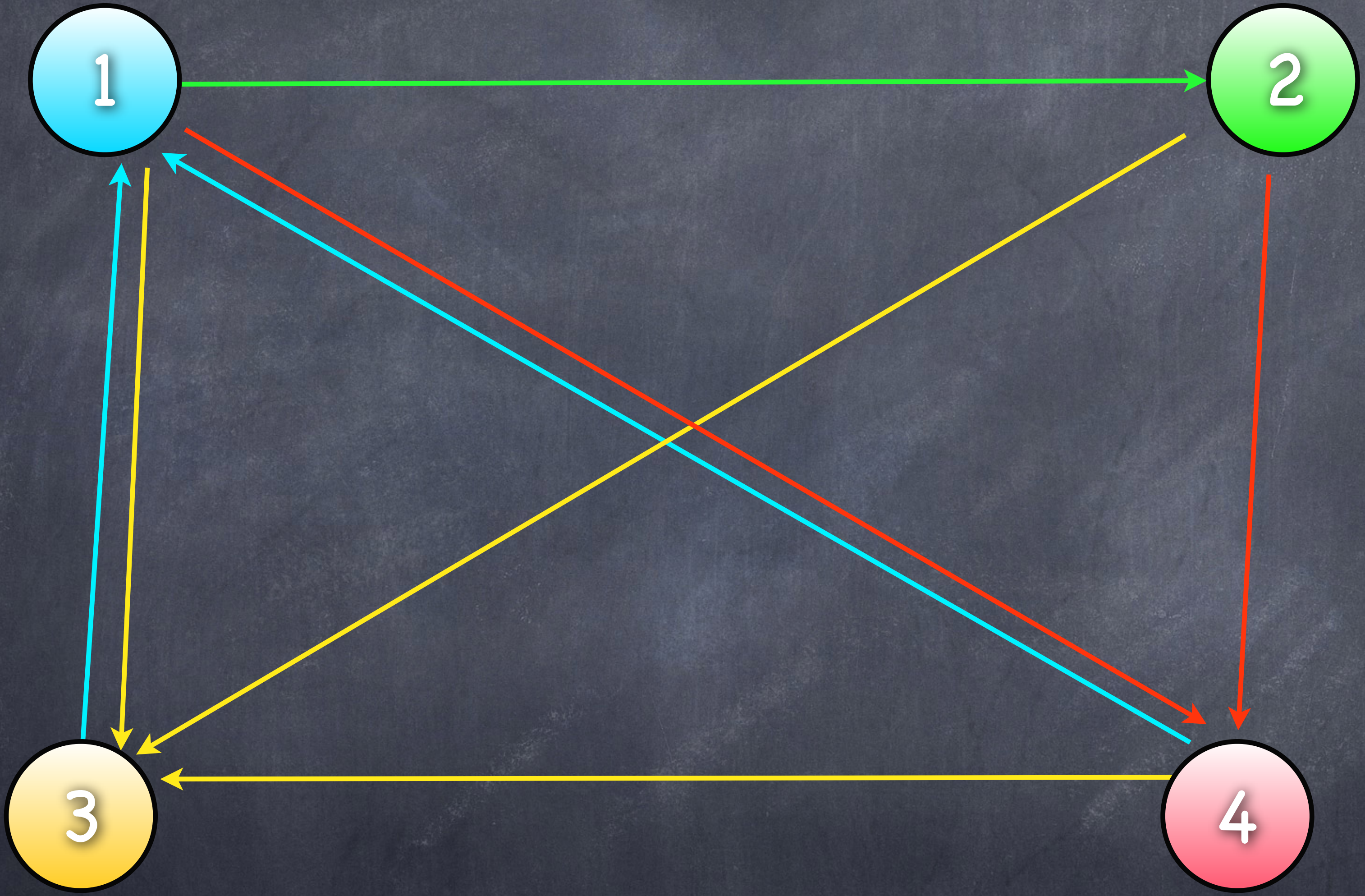
$Pr_i(j)$ = prob. at page j after i clicks

Random Surfer



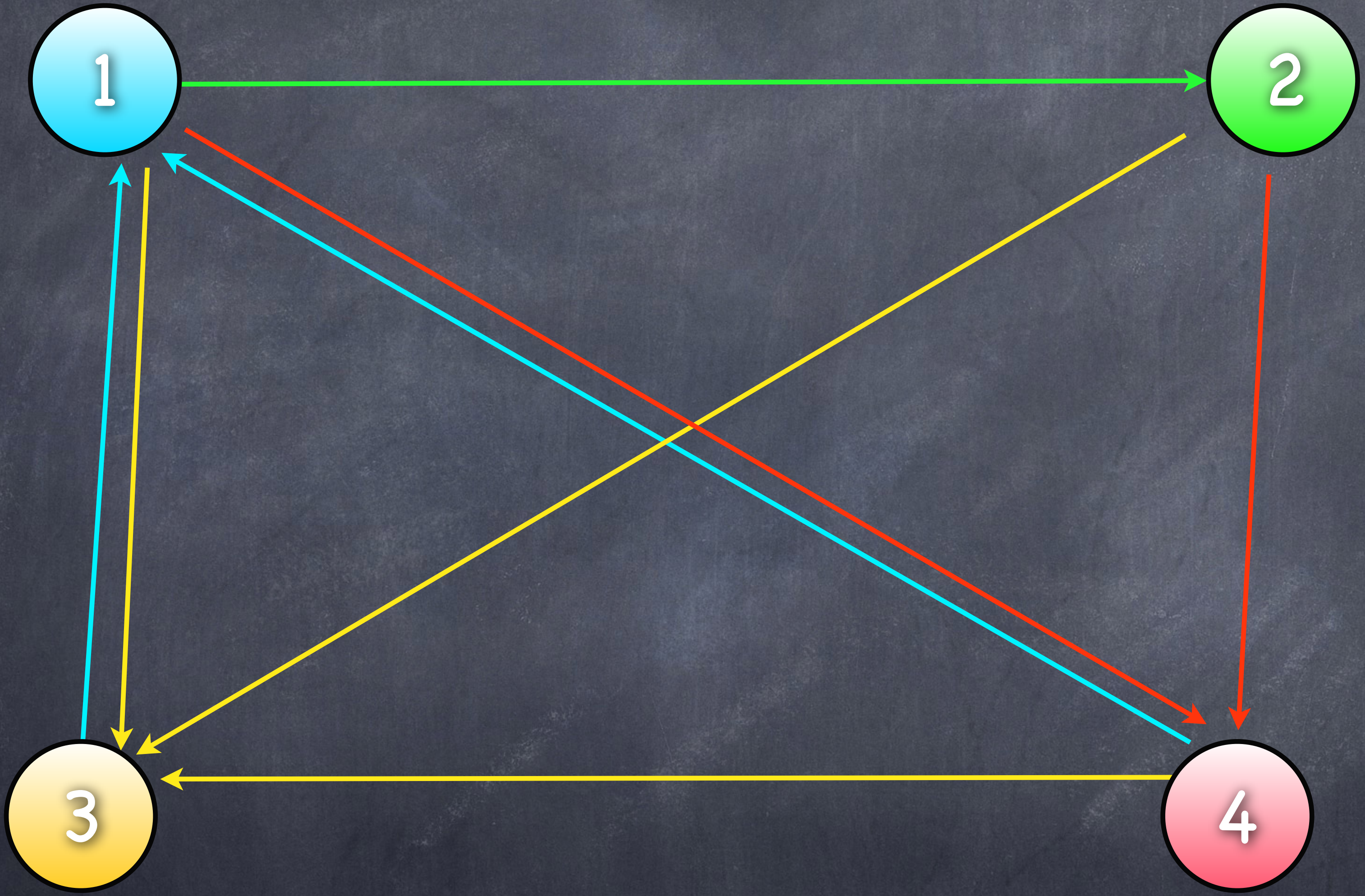
$$Pr_{i+1}(2) = Pr_i(1)/3$$

Random Surfer



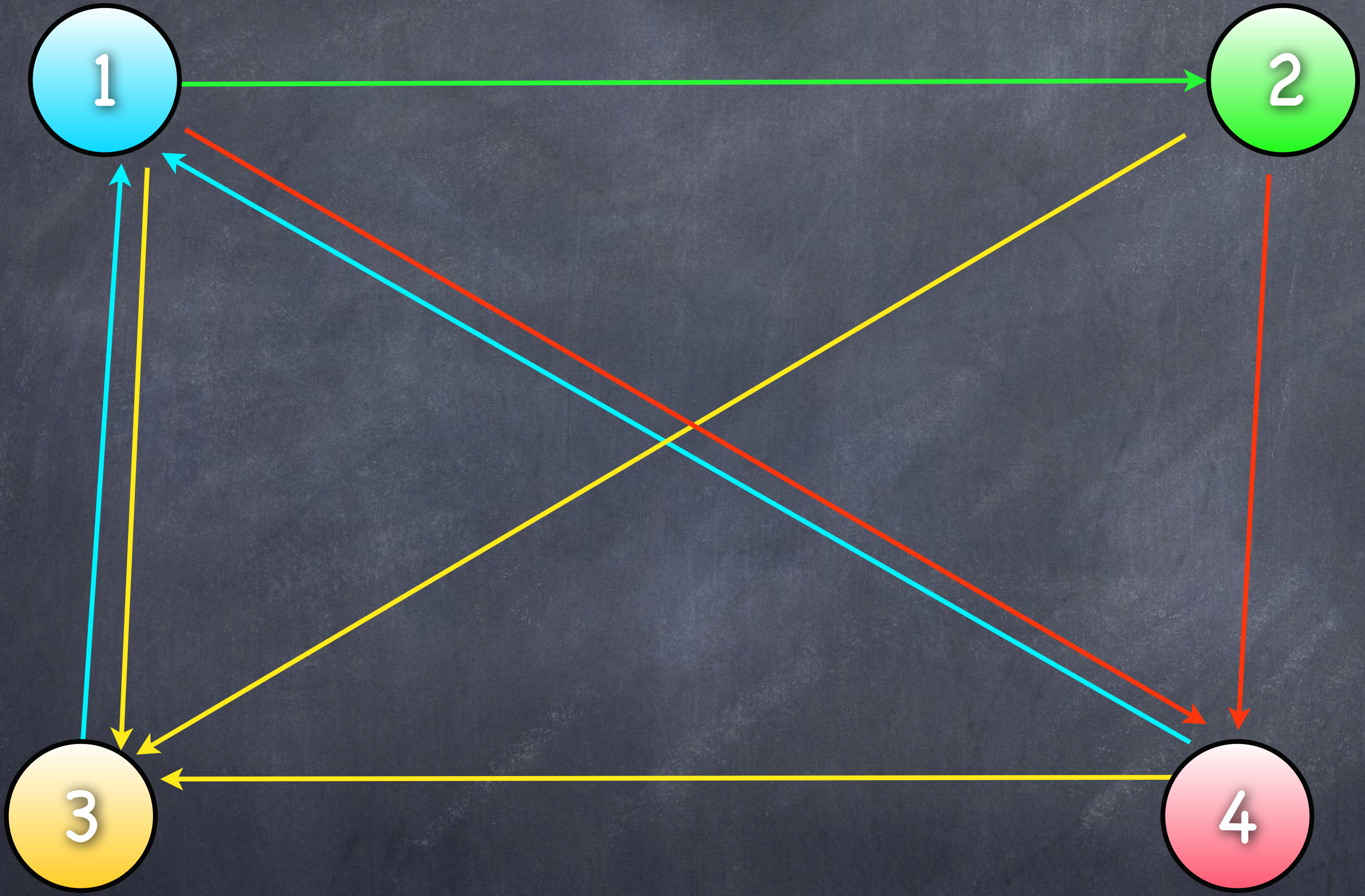
$Pr_{i+1}(1) = ?$

Random Surfer

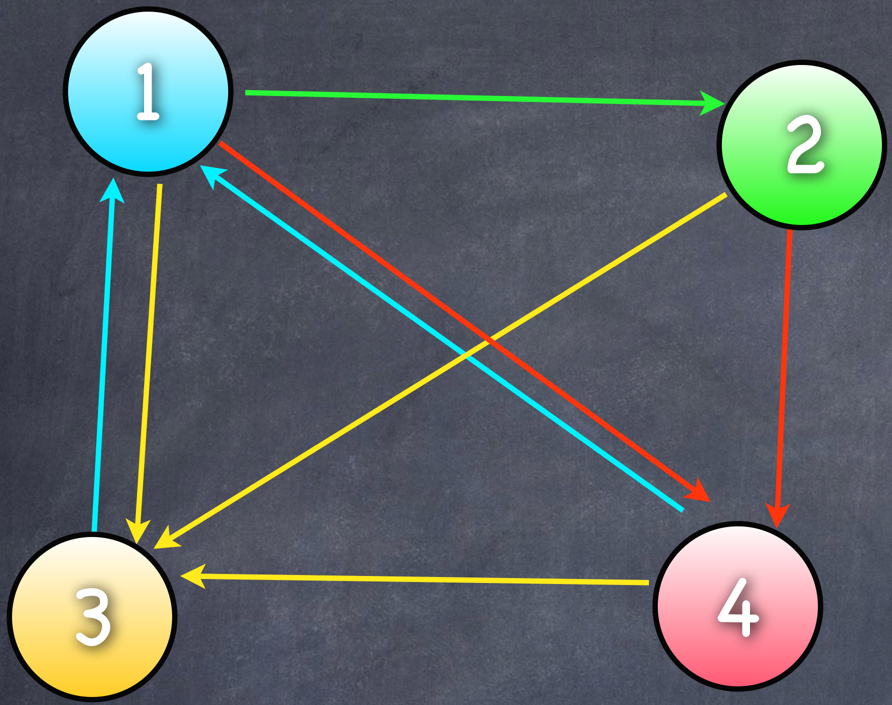


$$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$$

Random Surfer



$$Pr_{i+1}(j) = \sum_{k \in in(j)} Pr_i(k)/C(k)$$



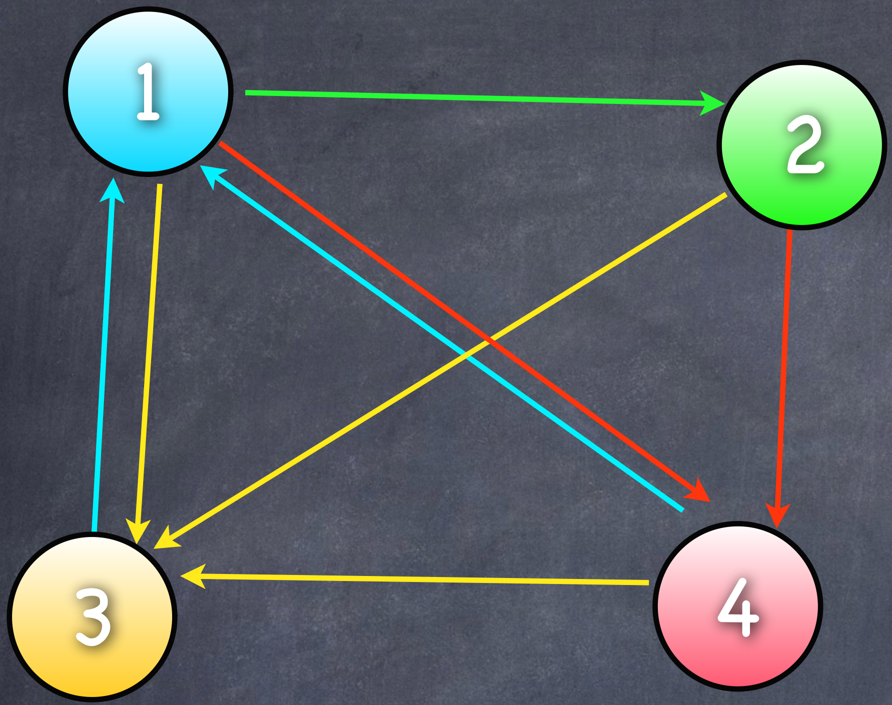
$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4			
2	1/4			
3	1/4			
4	1/4			



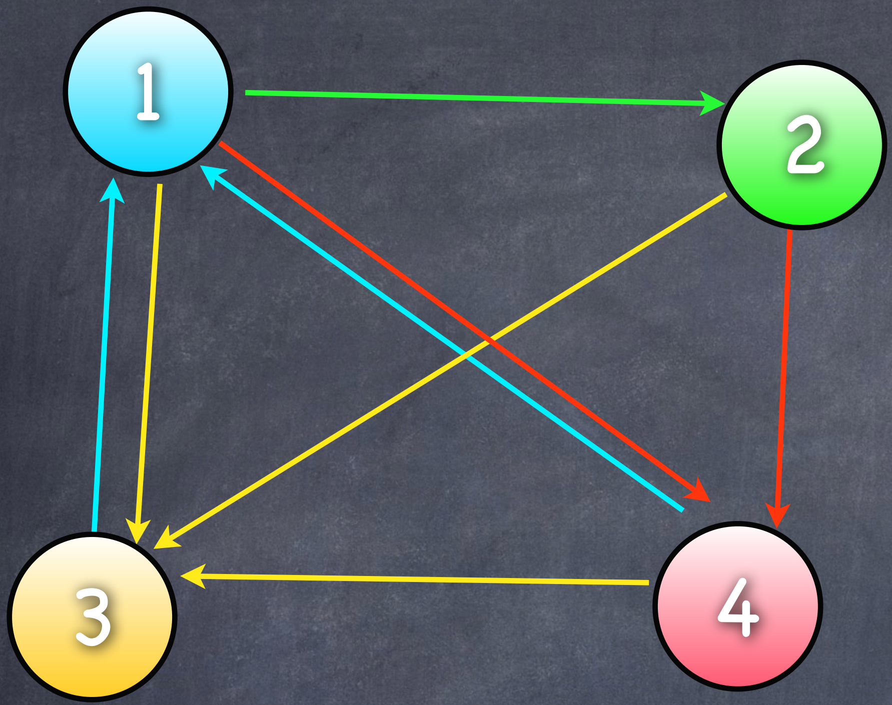
$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	3/8		
2	1/4			
3	1/4			
4	1/4			



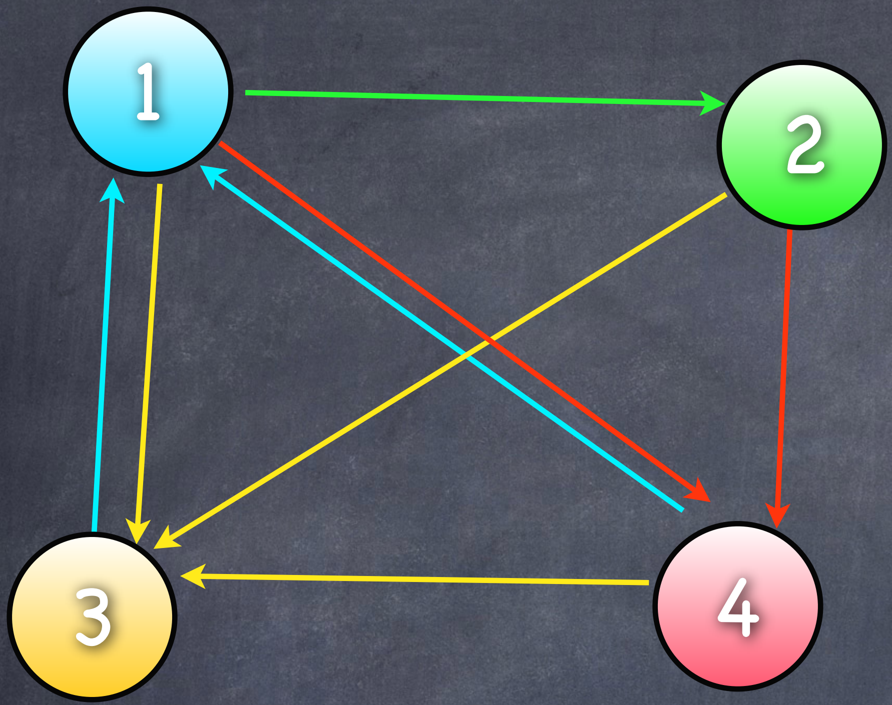
$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	3/8		
2	1/4	1/12		
3	1/4			
4	1/4			



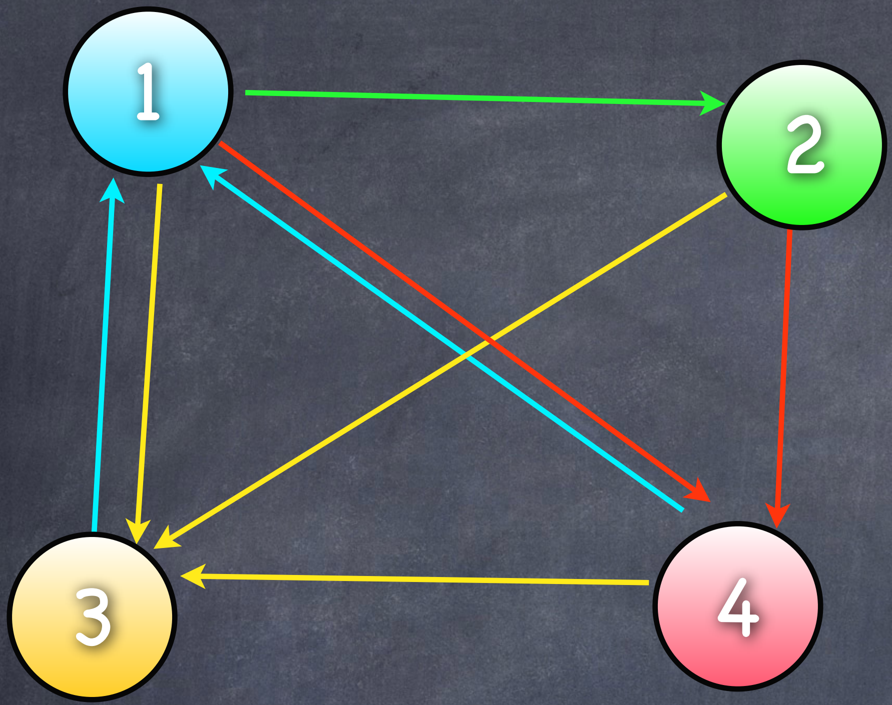
$$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$$

$$Pr_{i+1}(2) = Pr_i(1)/3$$

$$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$$

$$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	3/8		
2	1/4	1/12		
3	1/4	1/3		
4	1/4			



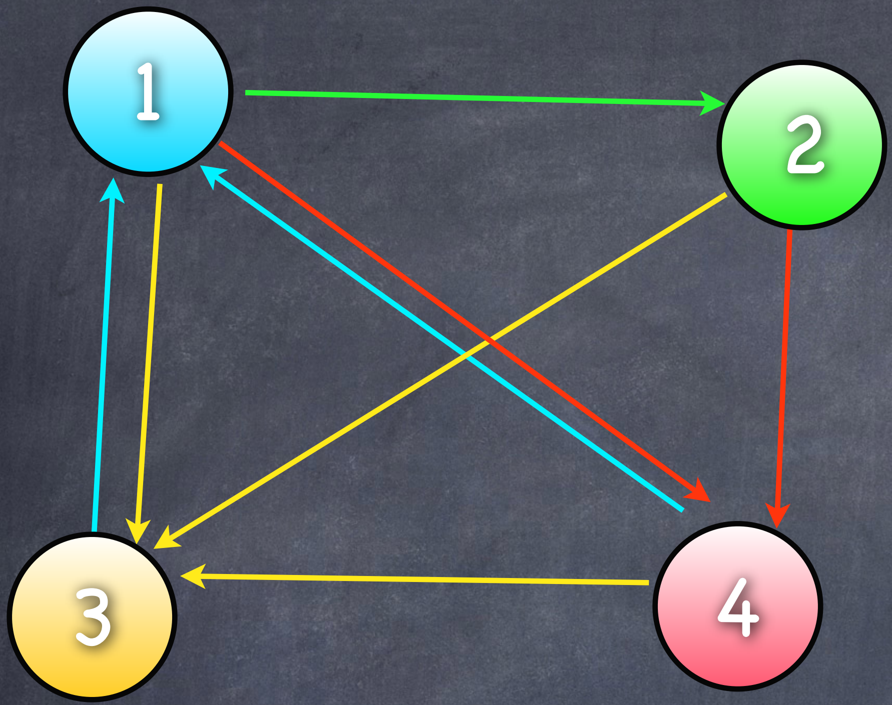
$$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$$

$$Pr_{i+1}(2) = Pr_i(1)/3$$

$$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$$

$$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	3/8		
2	1/4	1/12		
3	1/4	1/3		
4	1/4	5/24		



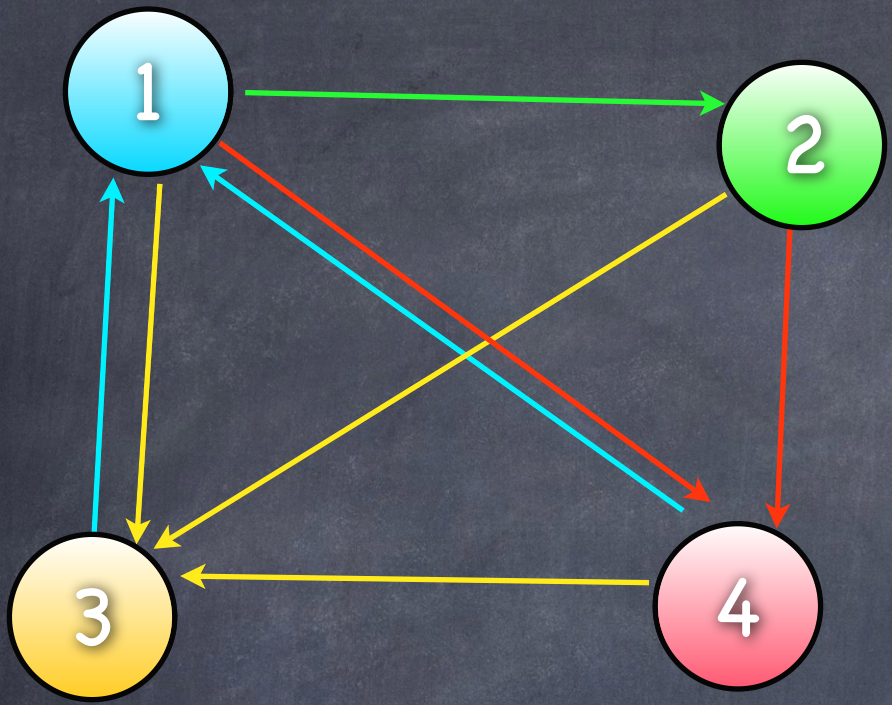
$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	0.38		
2	1/4	0.08		
3	1/4	0.33		
4	1/4	0.21		



$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	0.38	0.44	
2	1/4	0.08	0.54	
3	1/4	0.33	0.27	
4	1/4	0.21	0.17	

	$P_0()$	$P_1()$	$P_2()$	$P_3()$	$P_4()$	$P_5()$	$P_6()$	$P_7()$
1	0.25	0.38	0.44	0.35	0.40	0.39	0.38	0.39
2	0.25	0.08	0.13	0.15	0.12	0.13	0.13	0.13
3	0.25	0.33	0.27	0.29	0.30	0.29	0.29	0.29
4	0.25	0.21	0.17	0.21	0.19	0.19	0.20	0.19

What is Happening?

- The distributions $Pr_i()$ converge to a probability distribution $Pr_\infty()$
  - And it's the same regardless of starting distribution Pr_0!
  - $Pr_\infty()$ depends only on the **structure** of graph G
- How can we think about $Pr_\infty()$?

Understanding $Pr_\infty()$

- $Pr_\infty(v)$ is the probability of **eventually** being at vertex v after some **very long** random walk through the web graph, starting from a randomly selected vertex
- $Pr_\infty(v) = \Sigma_u Pr_\infty(u)/C(u)$ summing over all u→v
- $Pr_\infty()$ is called an equilibrium distribution for G
- If G is "properly structured", $Pr_\infty()$ exists and is unique!

Perron-Frobenius* Theorem

Let G be a **strongly connected and aperiodic**** directed graph and let $A_{v,u}$ be the probability of moving from vertex u to vertex v. Then there is a probability distribution Pr_∞ such that

- $Pr_\infty(v) = \Sigma_u Pr_\infty(u) * A_{v,u}$, summing over all $u \rightarrow v$
- Pr_∞ is the limit of Pr_0, Pr_1, Pr_2, Pr_3, ... : As $n \rightarrow \infty$, $Pr_n \rightarrow Pr_\infty$

Pr_∞ is called the equilibrium distribution and it's unique given A

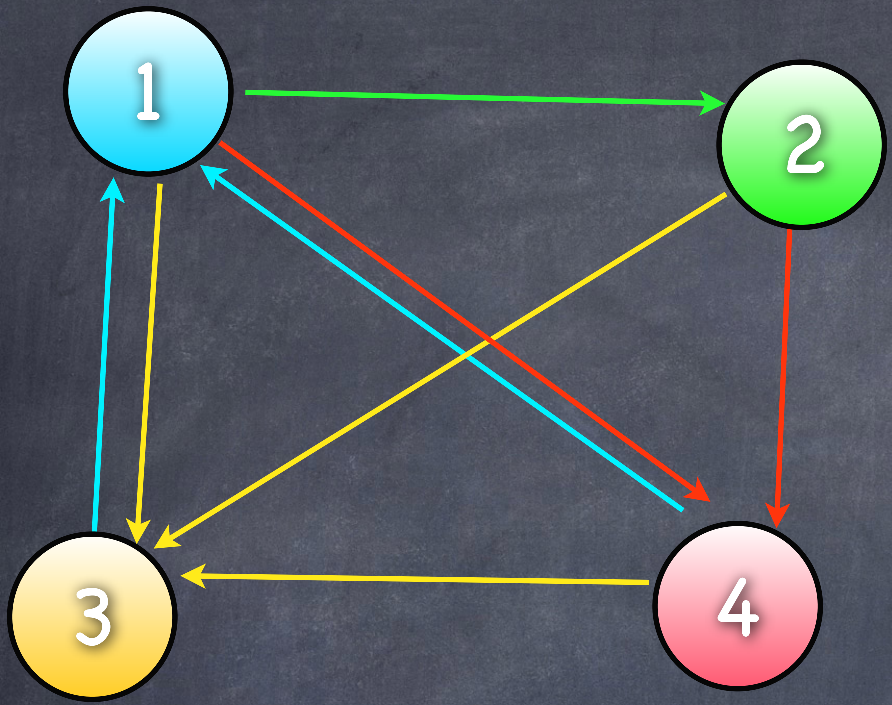
The fine print:

*This theorem describes a property of **matrices**. $A_{v,u}$ satisfies the hypotheses of the theorem and so $A_{v,u}$ has the property, which implies the existence of Pr_∞.

G is **k-periodic if the length of every cycle in G is a multiple of k > 1. If there is no such k, G is **aperiodic**. We can assume that the web graph is aperiodic.

What Could Go Wrong?

- The web graph is not strongly connected
    - There are pages with no links (sink)
    - There are groups of pages with no links leaving the group (connected component)



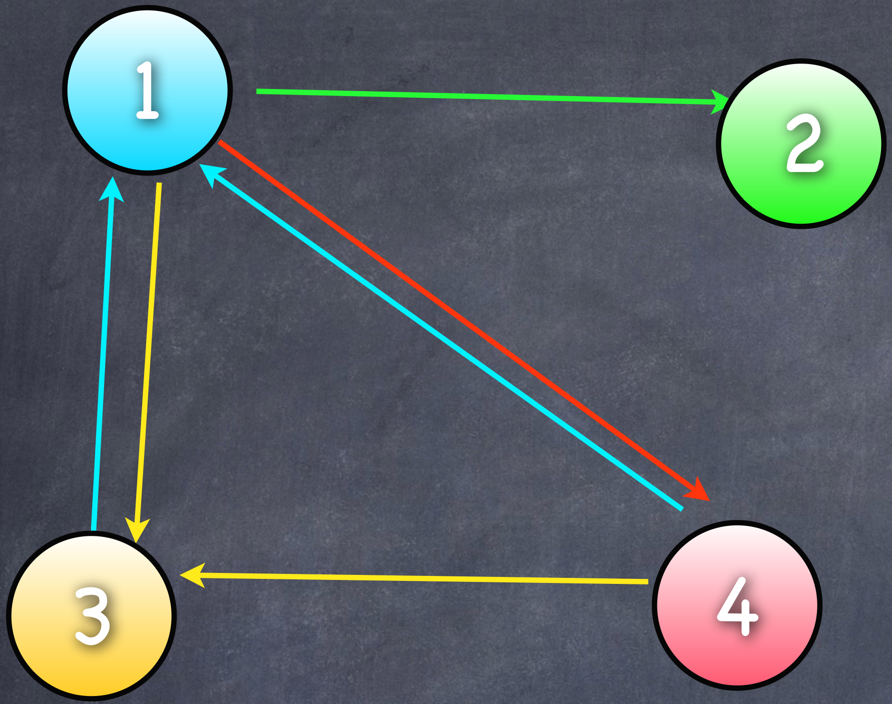
$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

$Pr_{i+1}(2) = Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(2)/2 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3 + Pr_i(2)/2$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	0.38	0.44	
2	1/4	0.08	0.13	
3	1/4	0.33	0.27	
4	1/4	0.21	0.17	



$Pr_{i+1}(1) = Pr_i(3) + Pr_i(4)/2$

Attention! $Pr_{i+1}(2) = Pr_i(2) + Pr_i(1)/3$

$Pr_{i+1}(3) = Pr_i(1)/3 + Pr_i(4)/2$

$Pr_{i+1}(4) = Pr_i(1)/3$

	$P_0()$	$P_1()$	$P_2()$	
1	1/4	0.38	0.25	0.23
2	1/4	0.33	0.46	0.54
3	1/4	0.21	0.17	0.15
4	1/4	0.08	0.13	0.08

Avoiding Traps

- The web graph is not strongly connected
  - There are pages with no links (sink)
  - There are groups of pages with no links leaving the group (connected component)
- What can we do?

Avoiding Traps

Random Walks : Jump!

- Adjust probabilities to allow for random page jumping
- Let E(v) be a probability distribution
    - Idea: E(v) = probability that user randomly jumped to page v from some other page

Avoiding Traps

Random Walks : Jump!

- $Pr_{i+1}(v) = \Sigma_u Pr_i(u)/C(u)$ (for u→v) becomes
- $Pr_{i+1}(v) = \delta E(v) + (1-\delta) \Sigma_u Pr_i(u)/C(u)$ (for u→v)
  - Why δ? : Ensure $Pr_{i+1}(v)$ forms a probability distribution (choose $\delta << 1$)
- Same as replacing $A_{v,u}$ with $\delta \cdot E(v) + (1 - \delta) \cdot A_{v,u}$
- Frobenius Theorem still holds : $Pr_\infty()$ exists

Avoiding Traps

Random Walks : Jump!

- Essentially, we've added all missing edges to the web graph, but given these new edges tiny probabilities
  - Probabilities of existing edges are also tweaked to ensure that we still have a probability distribution
  - Now graph is strongly connected and aperiodic (because it's complete)
  - The starting transition probabilities (matrix A) determine the equilibrium probabilities

Summary & Observations

- PageRank uses a combination of relevance and importance ranks
  - Relevance based on page (vertex) contents
  - Importance based on link structure (edges)
- Importance can be viewed as a probability distribution on the vertices (pages)