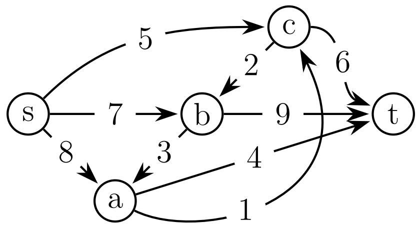
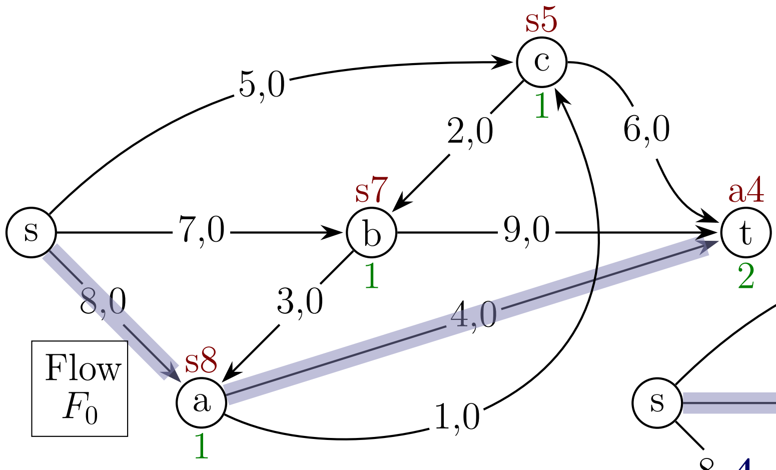
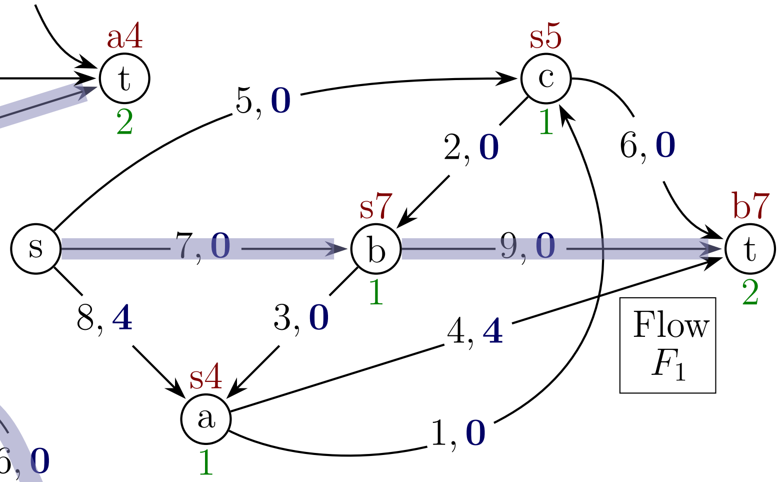
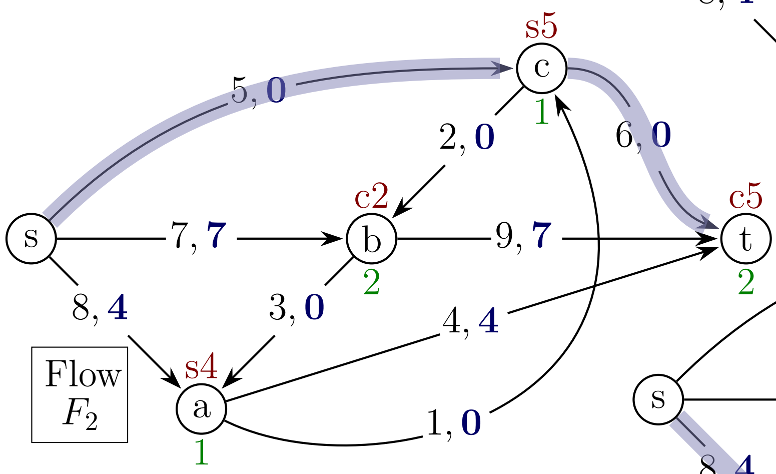
Question 1 The diagram at right shows the capacities and directions of all links in a network with source s, target t and intermediate nodes a, b and c. Use the labelling algorithm to find the maximum flow through the network and how it can be achieved. Prove that your flow is maximum by finding a cut of equal value.

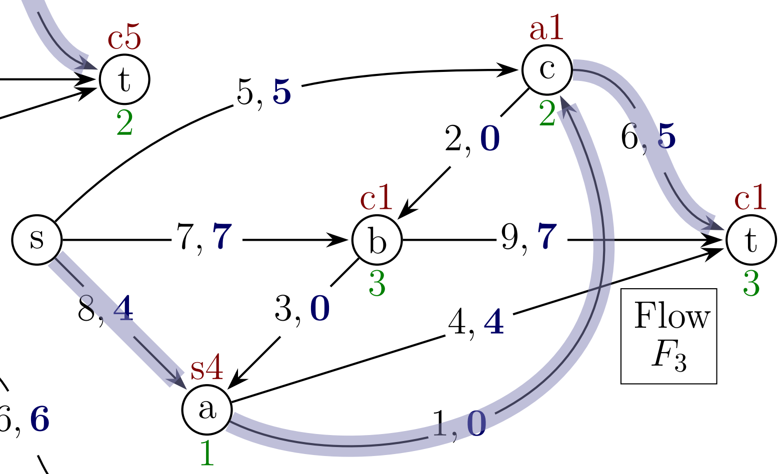


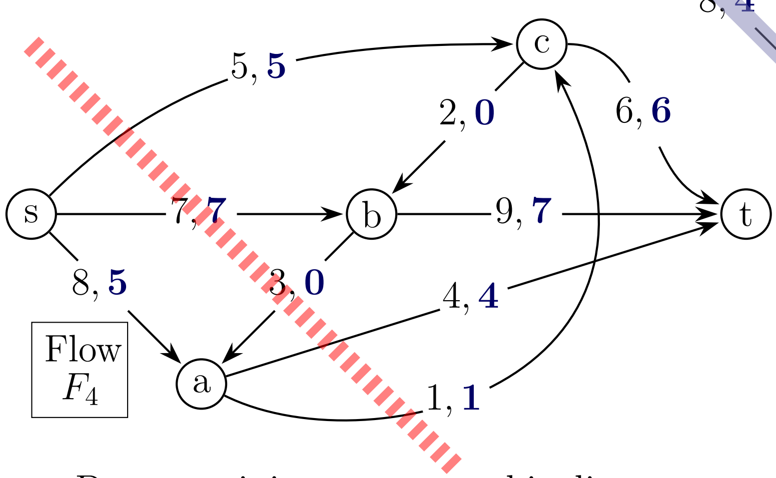
Here are some blank diagrams to fill in with **levels, labels** and **flows**. Use an outliner pen to mark the **incremental flow** dictated by the target's label. The first diagram is filled in as an example. You will need all diagrams. Draw a min cut on the last diagram.











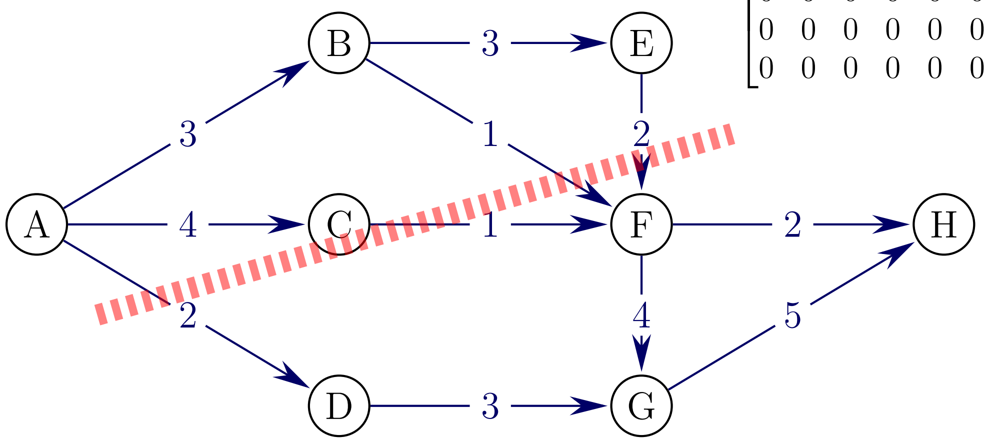
Draw a minimum cut on this diagram.

Max Flow Value = Min Cut Capacity = $\boxed{17}$

Question 2 The matrix at right shows the capacities for a network with source at vertex A and target at vertex H.

$$\begin{bmatrix} 0 & 3 & 4 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(a) Find a minimum cut. Specify the partition of the vertices, the edges making up the cut, and the value of the cut.

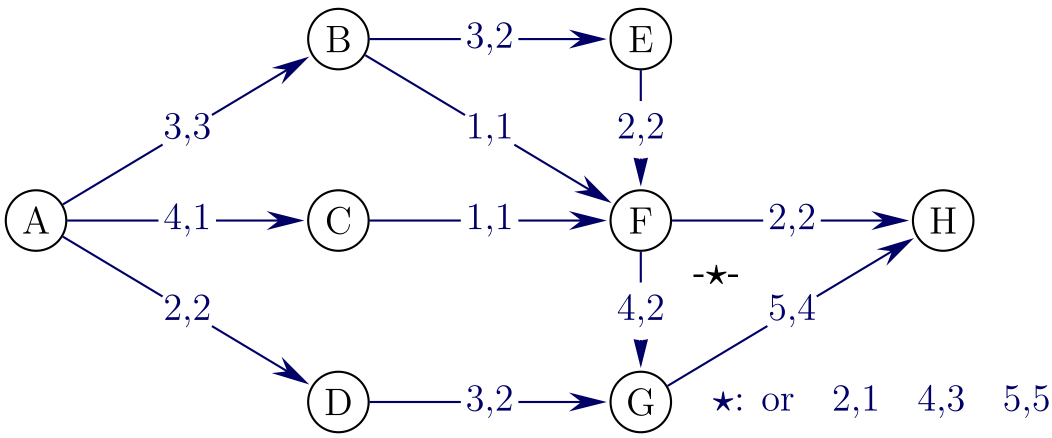


By inspection, after trying many possibilities, we find the cut indicated, comprising the edge set $\boxed{\{(A,D),(C,F),(B,F),(E,F)\}}$.

The corresponding partition of the vertex set is $\boxed{\{\{A,B,C,E\},\{D,F,G,H\}\}}$.

The capacity of the cut is $2+1+1+2=\boxed{6}$.

(b) Use the minimum cut to find a maximum flow, using only integer flows.
There is no need to use the labelling algorithm.
Draw a diagram showing capacity and flow value for every directed edge.



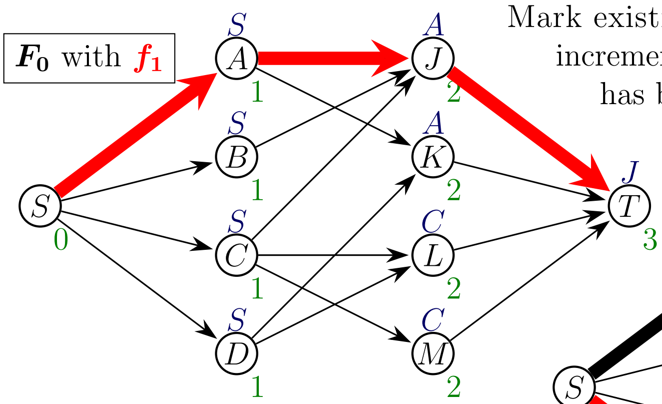
Note: The flows can be determined as follows:

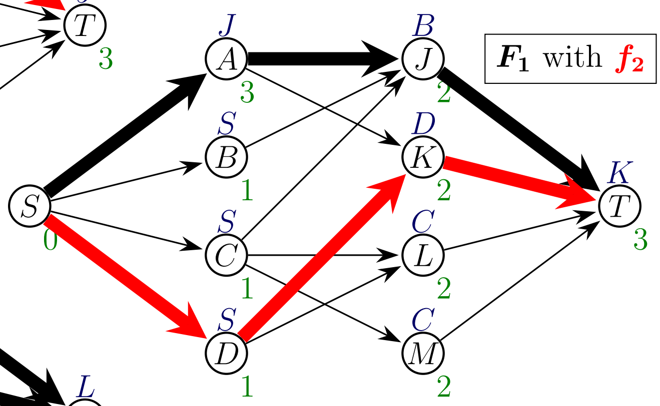
1. *A max flow must utilise the full capacity of the min cut, so we can mark on those flows first.*
2. *Backtracking from* E, B *and* C *gives three more.*
3. *Tracking forward from* D *gives one more.*
4. *There are just two ways to finish using integer flows.*

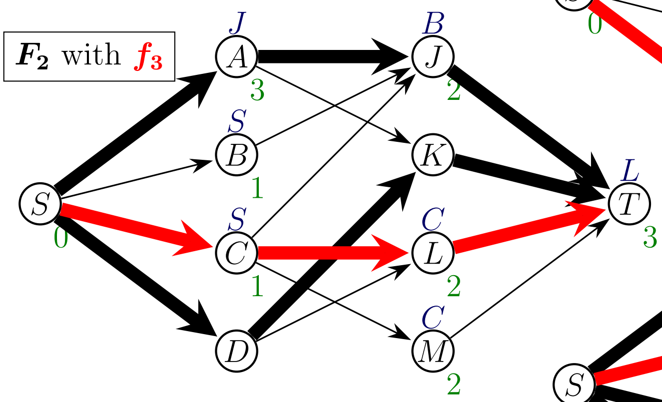
Question 3F our children at a daycare are selecting snuggle toys for nap time. Avril likes to snuggle the Jaguar and the Koala; Bai only likes to snuggle the Jaguar; Carletta like to snuggle the Jaguar, the Llama and the Moo-moo Cow; and Dan likes to snuggle the Koala and the Llama. Use the labelling algorithm to find a perfect match-up.

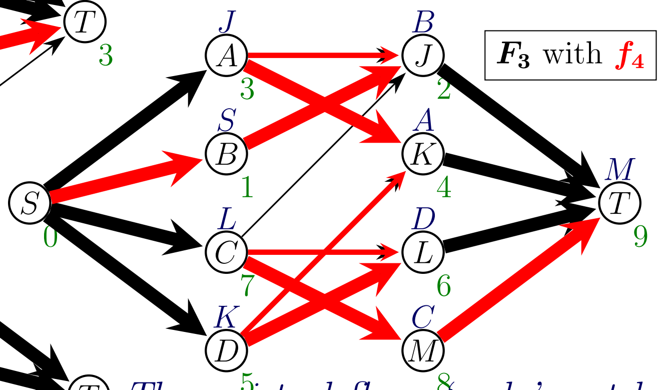
The diagrams show the relation "X **likes to snuggle** Y" from the set of **children** {A,B,C,D} to the set of possible **suggle toys** {J,K,L,M}. A 'supersource' S and a 'supertarget' T have been introduced. Step through from flow F_0 to F_4, showing all levels and labels.

Mark existing non-zero flows in heavy black and incremental flows in heavy red. The first diagram has been completed to indicate the style.

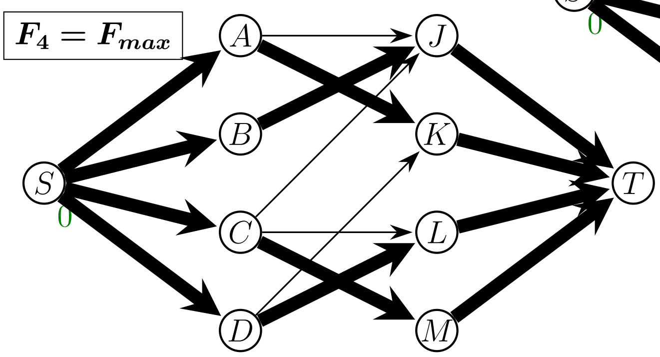




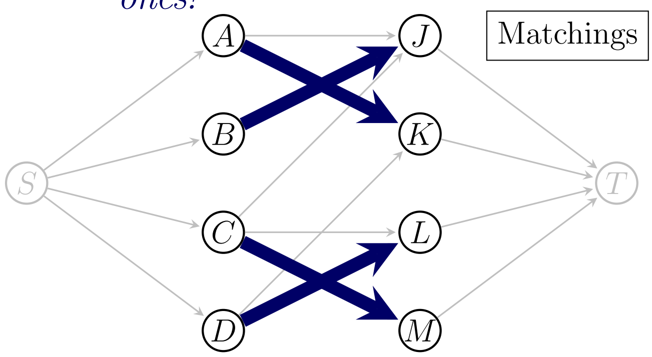




Three virtual flows 'undo' matchings made so far to create new ones!



Only **S** *has a level. In particular, no level is assigned to* **T**, *so the algorithm terminates.*

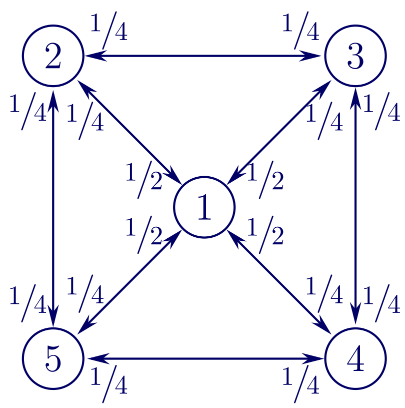


The matchings are the non-zero flows from **{A,B,C,D}** *to* **{J,K,L,M}**, i.e *Avril-Koala, Bai-Jaguar, Carletta-Moo-moo cow and Dan-Llama.*

Question 4 Let G be the digraph with adjacency matrix A shown at right. A random walker on G has all transition probabilities equal to $1/4$ except that all transition to vertex 1 have probability $1/2$.

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

(a) Draw the graph and mark on it the transition probabilities.



(b) Compile the transition matrix T and verify that it is stochastic.

$$T = \begin{bmatrix} 0 & 1/4 & 1/4 & 1/4 & 1/4 \\ 1/2 & 0 & 1/4 & 0 & 1/4 \\ 1/2 & 1/4 & 0 & 1/4 & 0 \\ 1/2 & 0 & 1/4 & 0 & 1/4 \\ 1/2 & 1/4 & 0 & 1/4 & 0 \end{bmatrix}.$$

All entries are non-negative and all rows sum to 1.
So the matrix is stochastic.

(c) On average over the long term, what proportion of time will the walker spend at vertex 1? Hint: There are really only two unknowns.

Let the required probability be p.

By symmetry the long term probabilities for all the other vertices will be equal, say q. *Then* $S = \begin{bmatrix} p & q & q & q & q \end{bmatrix}'$.

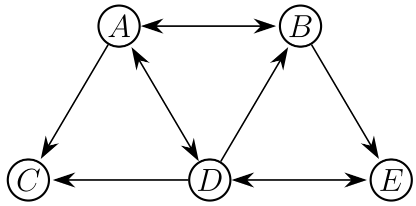
We need to solve $T'S = S$ *subject to* S *being a probability vector:*

$$\begin{aligned} \text{First row of } (T'-I)S=0 \quad &: \quad -p+\tfrac{1}{2}q+\tfrac{1}{2}q+\tfrac{1}{2}q+\tfrac{1}{2}q = 0^{1} \\ \text{Probability vector} \quad &: \quad p+q+q+q+q = 1 \\ \implies &\quad 6q = 1 \implies q = 1/6 \implies p = 1/3 \end{aligned}$$

So the walker will spend $\boxed{\textit{one third}}$ *of the time at vertex* 1.

[1] This is equivalent to Prob(in)=Prob(out) at vertex 1.

Question 5 For the webgraph shown at right, find the PageRanks of each page, assuming no damping. You will need to use the computer.



The PageRank algorithm assigns equal probabilities to out-links from each page, so if vertex i has out-degree n_i the probability of each of its out-links is $1/n_i$.

However if $n_i = 0$ the algorithm creates 'virtual' links to every other page (and even itself if $n \geq 10$) and assigns them equal probability.

$$\text{So} \quad T = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{bmatrix} 0 & 1/3 & 1/3 & 1/3 & 0 \\ 1/2 & 0 & 0 & 0 & 1/2 \\ 1/4 & 1/4 & 0 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 0 & 1/4 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array}.$$

Using $(T' - I)S = 0$ with 'shortcut' we need to solve:

$$\begin{bmatrix} -1 & 1/2 & 1/4 & 1/4 & 0 \\ 1/3 & -1 & 1/4 & 1/4 & 0 \\ 1/3 & 0 & -1 & 1/4 & 0 \\ 1/3 & 0 & 1/4 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} S = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

Equivalently, multiplying by 12 to get rid of fractions,

$$\begin{bmatrix} -12 & 6 & 3 & 3 & 0 \\ 4 & -12 & 3 & 3 & 0 \\ 4 & 0 & -12 & 3 & 0 \\ 4 & 0 & 3 & -12 & 12 \\ 12 & 12 & 12 & 12 & 12 \end{bmatrix} S = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 12 \end{bmatrix} \xRightarrow{\text{computer}^3} S = \begin{bmatrix} 0.195652 \\ 0.173913 \\ 0.139130 \\ 0.295652 \\ 0.195652 \end{bmatrix}.$$

So the approximate ranks of the pages are

Page:	*A*	*B*	*C*	*D*	*E*
PageRank:	*20%*	*17%*	*14%*	*30%*	*20%*

NOTE: The computer calculation can be carried out using Gaussian elimination in free online maths tools such as Matrix Reshish, MatrixCalc or WolframAlpha. Detailed examples of the use of the first two of these are provided in the solution to Q9 of the practice questions for section D3 of the course. All three tools allow the input of fractions, but entering integers is more convenient. Also, in the case of Matrix Reshish, fractional input forces fractional output, which is not helpful for this problem.

[3] I suggest Gauss-Jordan elimination in Matrix Reshish (`matrix.reshish.com`) for this.

Question 6 Let G be the graph with adjacency matrix A shown at right. Find the PageRank vector for this webgraph using a damping factor of 90%. You will need to use the computer.

$$A = \begin{bmatrix} 0&1&0&0&0&0&1&0&0&0\\ 0&0&0&1&1&0&0&0&0&1\\ 0&0&0&0&1&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&1\\ 1&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0&0&0\\ 1&0&0&1&0&1&0&0&1&0\\ 0&0&0&1&0&1&0&0&0&0\\ 0&1&0&0&0&1&1&0&0&0\\ 0&0&0&0&0&1&1&0&0&0 \end{bmatrix}$$

Step numbers below follow Example 5 in the lecture notes. Step 1 is not required.

Note that $\boldsymbol{n} = \mathbf{10}$ *so vertex 6 gets virtual links to all vertices including itself.*

Step 2:

$$\boldsymbol{T} = \begin{bmatrix} 0&1/2&0&0&0&0&1/2&0&0&0\\ 0&0&0&1/3&1/3&0&0&0&0&1/3\\ 0&0&0&0&1&0&0&0&0&0\\ 1/2&0&0&0&0&0&0&0&0&1/2\\ 1/2&0&0&0&0&0&0&0&0&1/2\\ 1/10&1/10&1/10&1/10&1/10&1/10&1/10&1/10&1/10&1/10\\ 1/4&0&0&1/4&0&1/4&0&0&1/4&0\\ 0&0&0&1/2&0&1/2&0&0&0&0\\ 0&1/3&0&0&0&1/3&1/3&0&0&0\\ 0&0&0&0&0&1/2&1/2&0&0&0 \end{bmatrix}$$

Step 3: *The damping factor is* $\mathbf{1-\boldsymbol{\alpha} = 90\% = 0.9}$. *So* $[\boldsymbol{I}-(\mathbf{1}-\boldsymbol{\alpha})\boldsymbol{T}']$

$$= \begin{bmatrix} 1&0&0&-.45&-.45&-.09&-.225&0&0&0\\ -.45&1&0&0&0&-.09&0&0&-.3&0\\ 0&0&1&0&0&-.09&0&0&0&0\\ 0&-.3&0&1&0&-.09&-.225&-.45&0&0\\ 0&-.3&-.9&0&1&-.09&0&0&0&0\\ 0&0&0&0&0&.91&-.225&-.45&-.3&-.45\\ -.45&0&0&0&0&-.09&1&0&-.3&-.45\\ 0&0&0&0&0&-.09&0&1&0&0\\ 0&0&0&0&0&-.09&-.225&0&1&0\\ 0&-.3&0&-.45&-.45&-.09&0&0&0&1 \end{bmatrix}$$

Step 4: $\dfrac{\boldsymbol{\alpha}}{\boldsymbol{n}} = \dfrac{.1}{\boldsymbol{n}} = .01$. *So* $\dfrac{\boldsymbol{\alpha}}{\boldsymbol{n}}\mathbf{1} = (.01\ .01\ \ldots\ .01)'$.

Step 5: *Using a computer*[4] *to solve* $[\boldsymbol{I}-(\mathbf{1}-\boldsymbol{\alpha})\boldsymbol{T}']\,\boldsymbol{PR} = \dfrac{\boldsymbol{\alpha}}{\boldsymbol{n}}\mathbf{1}$ *gives, to 3dp,*

$$\boldsymbol{PR} = (.143\ .107\ .024\ .104\ .077 .153\ .169\ .024\ .062\ .138)'.$$

As a check, we find the sum of the entries of $\boldsymbol{PR}$ *is 1, as it should be.*

[4] For decimal entry, I suggest 'Solving systems of linear equations' in MatrixCalc: `matrixcalc.org/en/`.