# Andrew_Goldberg_HW7

*Andrew Goldberg*

*3/14/2017*

**Page 304: #2**

The bridges and land masses of a certain city can be modeled with graph G in figure 8.7

a. Is G eulerian? Why or why not?
   No, vertexes 2,3,4,5 have degree 3, which is an odd number, whereas eulerian graphs require even degree
b. Suppose we relax the requiement of the walk so that the walker need not start and end at the sample land mass but still must travers every bridge exactly once. Is this type of wall possible in a city modeled by the graph? If so, how? If not, why not? Yes, 5-3-1-2-4-6-5-4-3-2

**Page 307: #1**

a. Write down the set of edges E(G)
   E(G) = {fe, fa, fd, ae, ab, bd, bc, cd, de}
b. Which edges are incident with vertex b?
   ba, bc, bd
c. Which vertices are adjacent to vertex c?
   b, d
d. Compute deg(a)
   3
e. Compute |E(G)|
   9

**Page 320: #10**

A basketball coach needs to find a starting lineup for her team. THere are five positions that must be filled: point guard (1), shooting guard (2), swing (3), power forward (4), and center (5). Given the data in table 8.7, create a graph model and use it to find a feasible starting lineup. What changes if the coach decides she can't play Hermione in position 3?

Couldn't get the graph to look like it does in textbook, but still works.

One feasible starting lineup:
1: Fay
2: Ellen
3: Hermione
4: Gladys
5: Deb

What changes if the coach decides she can't play Hermione in position 3?
Deb is the only one who can play position 5 (center), leaving gladys with position 4 (power forward), and hermoine the only one to play position 3 (swing). If hermoine can't play swing, then other options include leaving center or powerforward open while gladys or deb play swing. Personally, I'd keep the big girls in the power positions and fill the swing role with a shooting guard, who is typically taller than a point guard…

```
library(ggplot2)
library(network)
```

```
## network: Classes for Relational Data
## Version 1.13.0 created on 2015-08-31.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##                     Mark S. Handcock, University of California -- Los Angeles
##                     David R. Hunter, Penn State University
##                     Martina Morris, University of Washington
##                     Skye Bender-deMoll, University of Washington
##  For citation information, type citation("network").
##  Type help("network-package") to get started.
```

```
library(sna)
```

```
## Loading required package: statnet.common
```

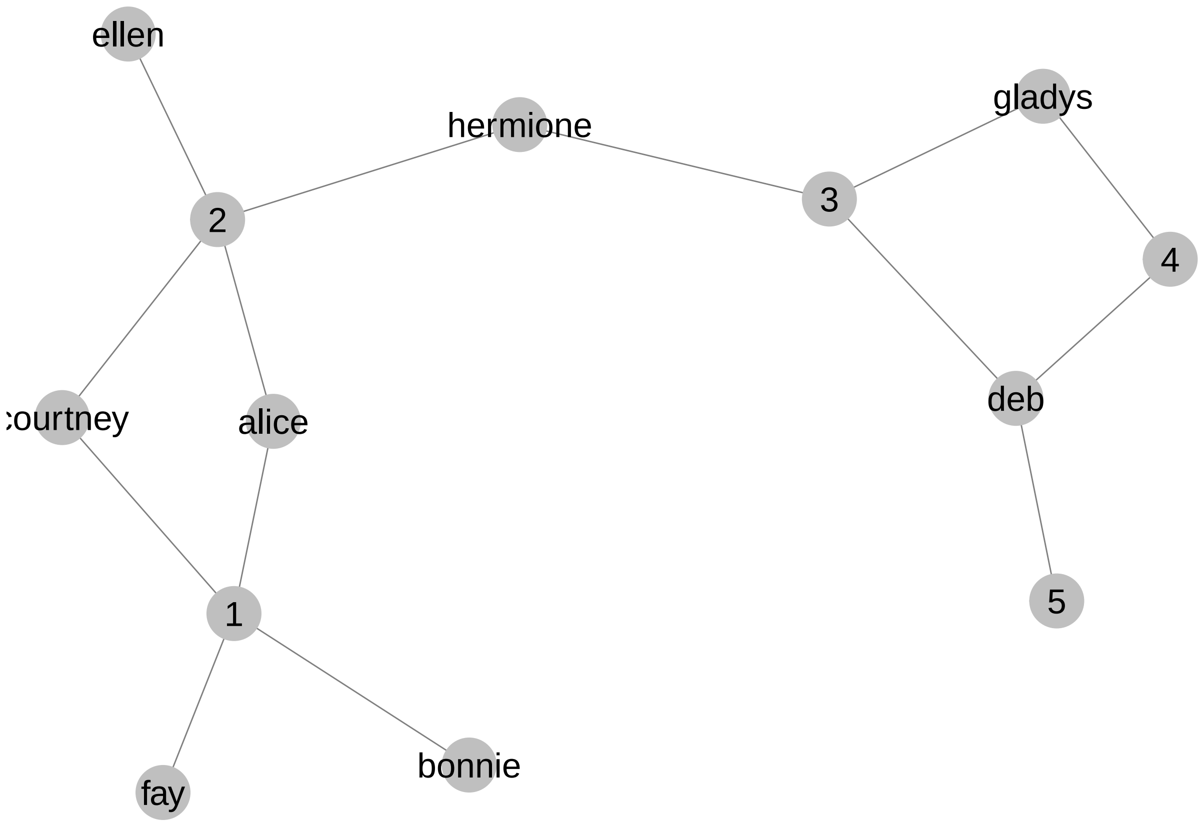
```
## sna: Tools for Social Network Analysis
## Version 2.4 created on 2016-07-23.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##  For citation information, type citation("sna").
##  Type help(package="sna") to get started.
```

```
library(GGally)
library(scales)

q320a <- data.frame(alice = c(1,1,0,0,0),
                   bonnie = c(1,0,0,0,0),
                   courtney = c(1,1,0,0,0),
                   deb = c(0,0,1,1,1),
                   ellen = c(0,1,0,0,0),
                   fay = c(1,0,0,0,0),
                   gladys = c(0,0,1,1,0),
                   hermione = c(0,1,1,0,0),
                   row.names = 1:5)

q320a <- network(q320a,
        matrix.type = "bipartite",
        ignore.eval = FALSE,
        names.eval = "weights")

ggnet2(q320a, label = T)
```




**Page 330: #1 (i'll assume we meant p331)**

Find the shortest path from node a to node j in the graph with edge weights shown on the graph. Using Dijkstra's shortest-path algorithm

L(v) = (L(a) + L(b) + L(c) + L(d) + L(e) + L(f) + L(g) + L(h) + L(i) + L(j)
1. (0, inf, inf, inf, inf, inf, inf, inf, inf, inf)
2. (0, 2, 4, inf, inf, inf, inf, inf, inf, inf)
3. (0, 2, 4, 4, 9, inf, inf, inf, inf, inf)
4. (0, 2, 4, 4, 8, 6, inf, inf, inf, inf)
5. (0, 2, 4, 4, 8, 6, 6, inf, inf, inf)
6. (0, 2, 4, 4, 8, 6, 6, inf, 12, inf)
7. (0, 2, 4, 4, 7, 6, 6, inf, 12, 14)
8. (0, 2, 4, 4, 7, 6, 6, 9, 10, 14)
9. (0, 2, 4, 4, 7, 6, 6, 9, 10, 13)
9. (0, 2, 4, 4, 7, 6, 6, 9, 10, 12)

So shortest path is a - b - d - g - e - i - j (12 edge weights)

**Page 330: #3**

Use our maximum-flow algorithm to find the maximum flow from s to t in the graph of figure 8.31 s-x1-y5-t = 1 (knocking out x1/y5)
s-x2-y6-t = 1 (knocking out x2/y6)
s-x3-y1-t = 1 (x4/y3)
s-x4-y3-1 = 1 (x4/y3)
So max flow is 4

**Page 338: #4**

Write down the linear program associated with solving maximum flow from s to t in the graph in figure 8.37.

Maximize z =

$$\sum_j x_{s,j}$$

(Which would be:)

$$x_{s,a}\ +\ x_{s,b}$$

Subject to:

$$\sum_i x_{i,j} = \sum_k x_{jk} \quad \forall j \in V(G) - s, t$$

(here:)

$$x_{s,a} = x_{a,c} + x_{a,b}$$

$$x_{a,b} + x_{s,b} = x_{b,c} + xb, d$$

$$x_{a,c} + x_{b,c} = x_{c,t} + x_{c,d}$$

$$x_{b,d} + x_{c,d} = x_{d,t}$$

$$x_{i,j} \leq u_{i,j} \quad \forall ij \in A(G)$$

(here:)

$$x_{s,a} \leq 3$$

$$x_{s,b} \leq 5$$

$$x_{a,c} \leq 6$$

$$x_{a,b} \leq 2$$

$$x_{c,b} \leq 2$$

$$x_{b,d} \leq 4$$

$$x_{c,d} \leq 1$$

$$x_{c,t} \leq 4$$

$$x_{d,t} \leq 5$$

***

$$x_{i,j} \leq 0 \quad \forall ij \in A(G)$$

(here:)

$$x_{s,a} \geq 0$$

$$x_{s,b} \geq 0$$

$$x_{a,c} \geq 0$$

$$x_{a,b} \geq 0$$

$$x_{c,b} \geq 0$$

$$x_{b,d} \geq 0$$

$$x_{c,d} \geq 0$$

$$x_{c,t} \geq 0$$

$$x_{d,t} \geq 0$$