Exercises Week 13

Aaron Palumbo

November 17, 2015

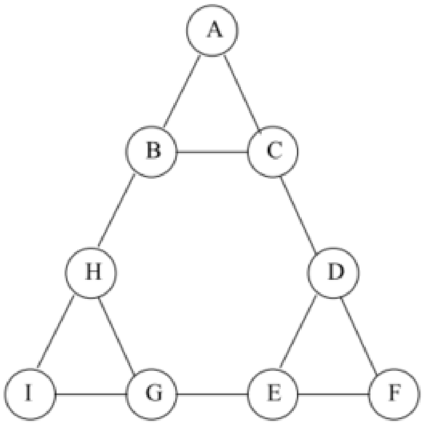
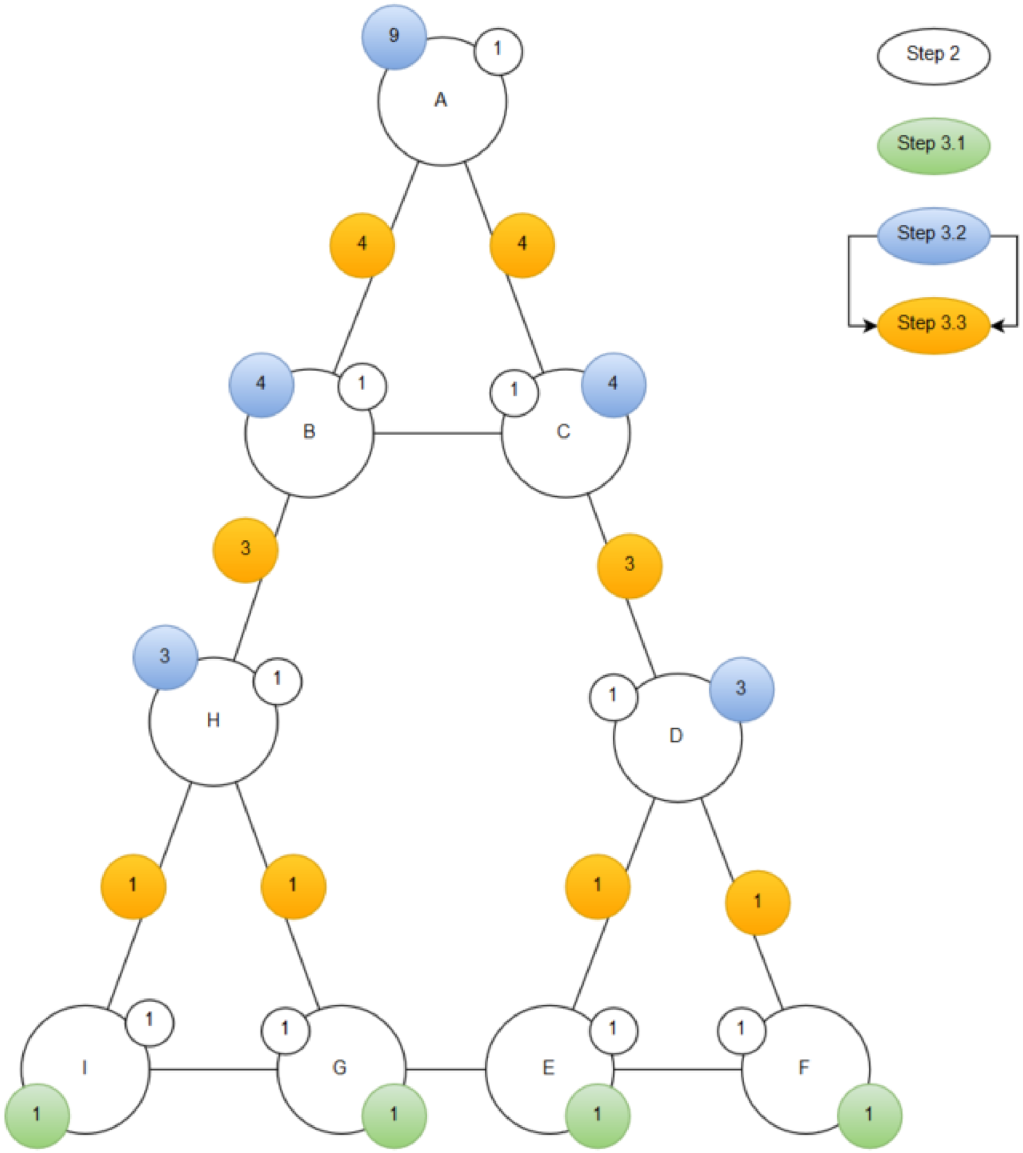


Figure 10.9: Graph for exercises

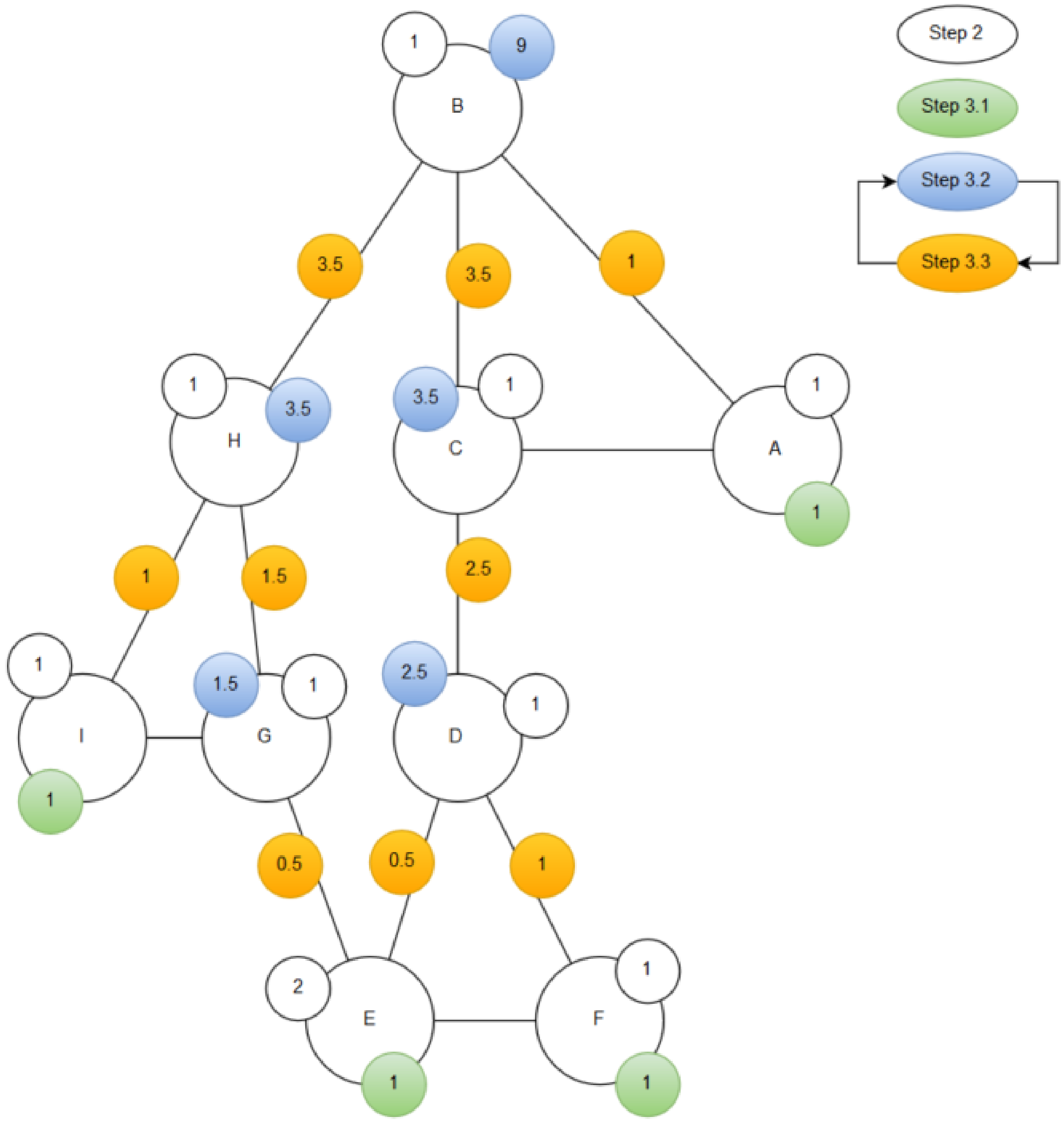
10.2.1 (section 10.2.6)

Figure 10.9 is an example of a social-network graph. Use the Girvan-Newman approach to find the number of shortest paths from each of the following odes that pass through each of the edges.

10.2.1 (a): A

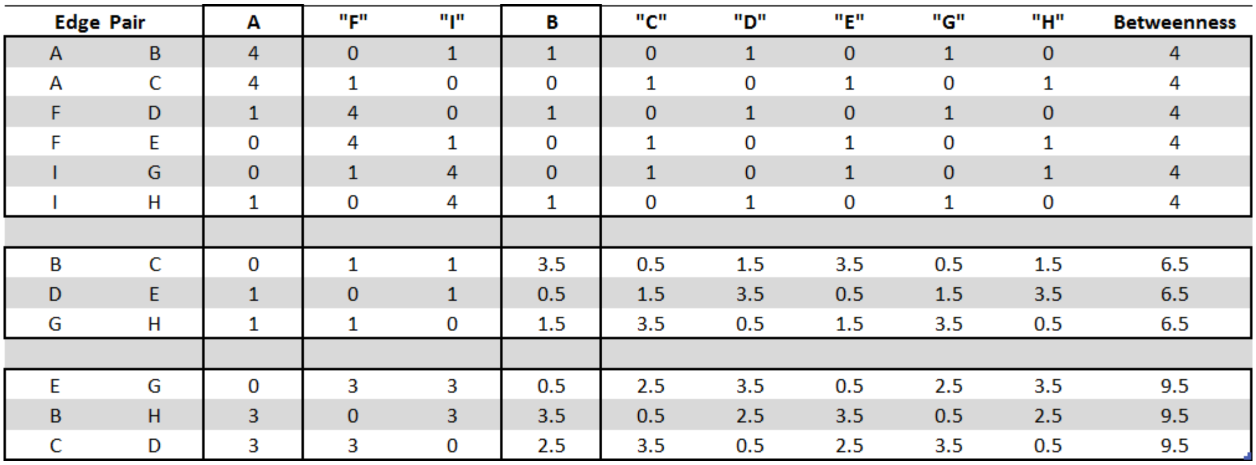

9
1
A
Step 2
Step 3.1
Step 3.2
Step 3.3
4
4
4
1
B
1
4
C
3
3
3
1
H
1
3
D
1
1
1
1
1
I
1
1
G
1
1
E
1
1
F
1
1

10.2.1 (b): B


1
9
B
Step 2
Step 3.1
Step 3.2
Step 3.3
3.5
3.5
1
1
3.5
H
3.5
1
C
1
A
1
1
1.5
2.5
1
I
1
1.5
1
G
2.5
D
1
0.5
0.5
1
2
E
1
1
F
1

10.2.2 (section 10.2.6)

Using symmetry, the calculations of Exercise 10.2.1 are all you need to compute the betweenness of each edge. Do the calculation.



Edge Pair		A	"F"	"I"	B	"C"	"D"	"E"	"G"	"H"	Betweenness
A	B	4	0	1	1	0	1	0	1	0	4
A	C	4	1	0	0	1	0	1	0	1	4
F	D	1	4	0	1	0	1	0	1	0	4
F	E	0	4	1	0	1	0	1	0	1	4
I	G	0	1	4	0	1	0	1	0	1	4
I	H	1	0	4	1	0	1	0	1	0	4
B	C	0	1	1	3.5	0.5	1.5	3.5	0.5	1.5	6.5
D	E	1	0	1	0.5	1.5	3.5	0.5	1.5	3.5	6.5
G	H	1	1	0	1.5	3.5	0.5	1.5	3.5	0.5	6.5
E	G	0	3	3	0.5	2.5	3.5	0.5	2.5	3.5	9.5
B	H	3	0	3	3.5	0.5	2.5	3.5	0.5	2.5	9.5
C	D	3	3	0	2.5	3.5	0.5	2.5	3.5	0.5	9.5

There are three groups of edges whose members, because of the symmetry of the network, will have the same betweenness. The actual calculations are shown for root nodes A and B. The rest of the root nodes are shown as place holders. The values in the place holder columns may not be exactly correct but taken together (again due to symmetry) the betweenness score for each edge is correct.

10.4.1 (section 10.4.6)

For the graph of Fig. 10.9, construct:

10.4.1 (a)

The adjacency matrix.

```
                   #A, B, C, D, E, F, G, H, I
adj_mat <- matrix(c(0, 1, 1, 0, 0, 0, 0, 0, 0,  #A
                    1, 0, 1, 0, 0, 0, 0, 1, 0,  #B
                    1, 1, 0, 1, 0, 0, 0, 0, 0,  #C
                    0, 0, 1, 0, 1, 1, 0, 0, 0,  #D
                    0, 0, 0, 1, 0, 1, 1, 0, 0,  #E
                    0, 0, 0, 1, 1, 0, 0, 0, 0,  #F
                    0, 0, 0, 0, 1, 0, 0, 1, 1,  #G
                    0, 1, 0, 0, 0, 0, 1, 0, 1,  #H
                    0, 0, 0, 0, 0, 0, 1, 1, 0), #I
                  nrow = 9, byrow = TRUE)

nodes <- unlist(strsplit("ABCDEFGHI", ""))
colnames(adj_mat) <- nodes
rownames(adj_mat) <- nodes

knitr::kable(adj_mat)
```

	A	B	C	D	E	F	G	H	I
A	0	1	1	0	0	0	0	0	0
B	1	0	1	0	0	0	0	1	0
C	1	1	0	1	0	0	0	0	0
D	0	0	1	0	1	1	0	0	0
E	0	0	0	1	0	1	1	0	0
F	0	0	0	1	1	0	0	0	0
G	0	0	0	0	1	0	0	1	1
H	0	1	0	0	0	0	1	0	1
I	0	0	0	0	0	0	1	1	0

10.4.1 (b)

The degree matrix.

```
                   #A, B, C, D, E, F, G, H, I
deg_mat <- matrix(c(2, 0, 0, 0, 0, 0, 0, 0, 0,  #A
                    0, 3, 0, 0, 0, 0, 0, 0, 0,  #B
                    0, 0, 3, 0, 0, 0, 0, 0, 0,  #C
                    0, 0, 0, 3, 0, 0, 0, 0, 0,  #D
                    0, 0, 0, 0, 3, 0, 0, 0, 0,  #E
                    0, 0, 0, 0, 0, 2, 0, 0, 0,  #F
                    0, 0, 0, 0, 0, 0, 3, 0, 0,  #G
                    0, 0, 0, 0, 0, 0, 0, 3, 0,  #H
                    0, 0, 0, 0, 0, 0, 0, 0, 2), #I
```

```
                  nrow = 9, byrow = TRUE)

colnames(deg_mat) <- nodes
rownames(deg_mat) <- nodes

knitr::kable(deg_mat)
```

	A	B	C	D	E	F	G	H	I
A	2	0	0	0	0	0	0	0	0
B	0	3	0	0	0	0	0	0	0
C	0	0	3	0	0	0	0	0	0
D	0	0	0	3	0	0	0	0	0
E	0	0	0	0	3	0	0	0	0
F	0	0	0	0	0	2	0	0	0
G	0	0	0	0	0	0	3	0	0
H	0	0	0	0	0	0	0	3	0
I	0	0	0	0	0	0	0	0	2

10.4.1 (c)

The Laplacian matrix.

```
lap_mat <- deg_mat - adj_mat

colnames(lap_mat) <- nodes
rownames(lap_mat) <- nodes

knitr::kable(lap_mat)
```

	A	B	C	D	E	F	G	H	I
A	2	-1	-1	0	0	0	0	0	0
B	-1	3	-1	0	0	0	0	-1	0
C	-1	-1	3	-1	0	0	0	0	0
D	0	0	-1	3	-1	-1	0	0	0
E	0	0	0	-1	3	-1	-1	0	0
F	0	0	0	-1	-1	2	0	0	0
G	0	0	0	0	-1	0	3	-1	-1
H	0	-1	0	0	0	0	-1	3	-1
I	0	0	0	0	0	0	-1	-1	2