

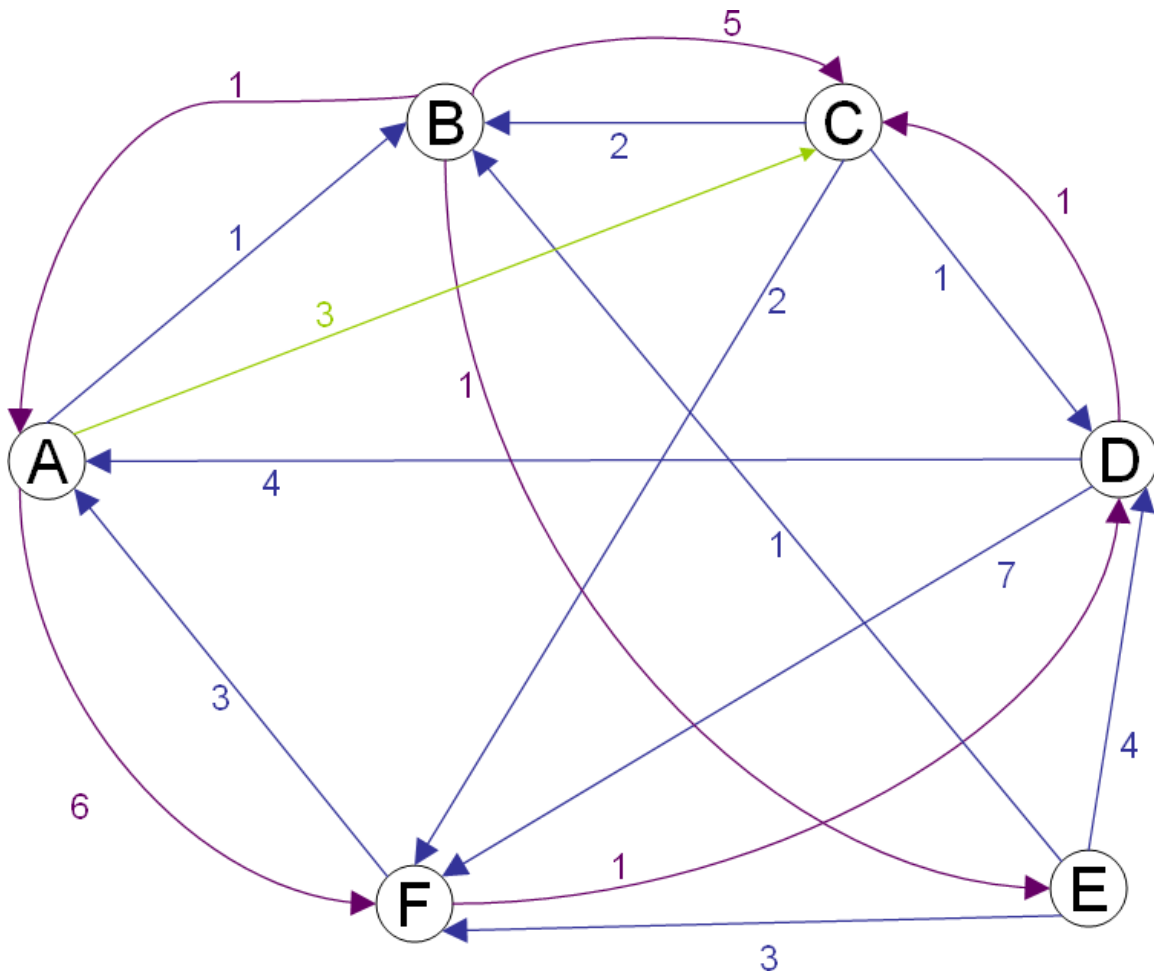
Course ISE 435: Distributed Algorithms in Network  
Communication  
Recitation 10 Exercise

Michael J. May

May 26, 2010

## 1 Floyd-Warshall Algorithm

For this recitation we will calculate the Floyd-Warshall algorithm on the graph shown below:



We will perform the Floyd-Warshall algorithm iteratively by including the nodes in alphabetical order.

## 2 Step 1: $S = \emptyset$

We begin with  $S = \emptyset$ , so the table is just the neighbor distances:

	A	B	C	D	E	F
A	0	1	3	Inf	Inf	6
B	1	0	5	Inf	1	Inf
C	Inf	2	0	1	Inf	2
D	4	Inf	1	0	Inf	7
E	Inf	1	Inf	4	0	3
F	3	Inf	Inf	1	Inf	0

## 3 Step 2: $S = \{A\}$

Adding  $A$  to  $S$ , we now must examine all paths that begin with  $A$  to see if they are better than ones that we already know. There is no need to examine the column or row  $A$  since they cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	Inf	Inf	6
B	1	0	4	Inf	1	7
C	Inf	2	0	1	Inf	2
D	4	5	1	0	Inf	7
E	Inf	1	Inf	4	0	3
F	3	4	6	1	Inf	0

## 4 Step 3: $S = \{A, B\}$

Adding  $B$  to  $S$ , we now must examine all paths that begin with  $B$  to see if they are better than the ones that we already know. There is no need to examine the row or column  $B$  since they are cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	Inf	2	6
B	1	0	4	Inf	1	7
C	3	2	0	1	3	2
D	4	5	1	0	6	7
E	2	1	5	4	0	3
F	3	4	6	1	5	0

## 5 Step 4: $S = \{A, B, C\}$

Adding  $C$  to  $S$ , we now must examine all paths that begin with  $C$  to see if they are better than the ones that we already know. There is no need to examine the row or column  $C$  since they are cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	4	2	5
B	1	0	4	5	1	6
C	3	2	0	1	3	2
D	4	3	1	0	4	3
E	2	1	5	4	0	3
F	3	4	6	1	5	0

## 6 Step 5: $S = \{A, B, C, D\}$

Adding  $D$  to  $S$ , we now must examine all paths that begin with  $D$  to see if they are better than the ones that we already know. There is no need to examine the row or column  $D$  since they are cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	4	2	5
B	1	0	4	5	1	6
C	3	2	0	1	3	2
D	4	3	1	0	4	3
E	2	1	5	4	0	3
F	3	4	2	1	5	0

## 7 Step 6: $S = \{A, B, C, D, E\}$

Adding  $E$  to  $S$ , we now must examine all paths that begin with  $E$  to see if they are better than the ones that we already know. There is no need to examine the row or column  $E$  since they are cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	4	2	5
B	1	0	4	5	1	4
C	3	2	0	1	3	2
D	4	3	1	0	4	3
E	2	1	5	4	0	3
F	3	4	2	1	5	0

## 8 Step 7: $S = \{A, B, C, D, E, F\}$

Adding  $F$  to  $S$ , we now must examine all paths that begin with  $F$  to see if they are better than the ones that we already know. There is no need to examine the row or column  $F$  since they are cannot change in this round.

	A	B	C	D	E	F
A	0	1	3	4	2	5
B	1	0	4	5	1	4
C	3	2	0	1	3	2
D	4	3	1	0	4	3
E	2	1	5	4	0	3
F	3	4	2	1	5	0

## 9 Conclusion

Having now added all of the nodes, we have arrived at the all-points shortest distance table. Adapting this algorithm to include the next hop is simple, just note the round during which any value was changed.