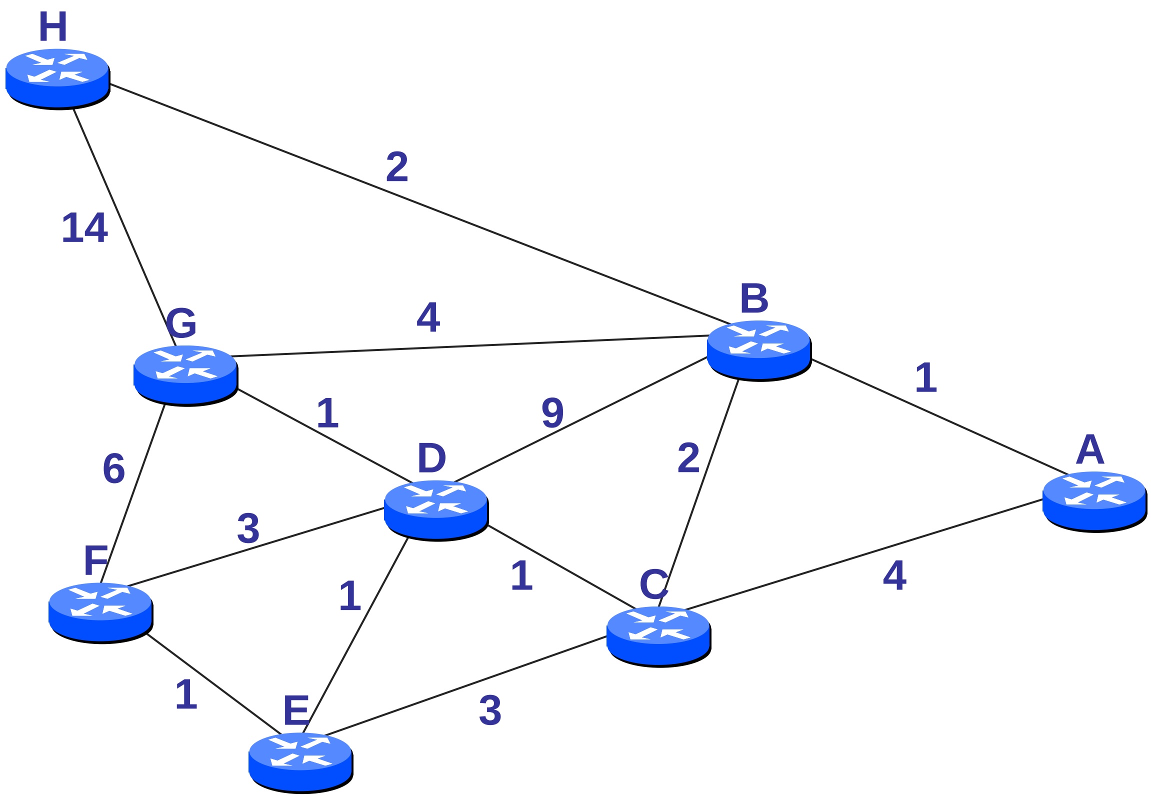
CSE 123a Winter 2005 Homework #2
Instructor: Stefan Savage
Homework TA: David Yu (davidyu@ucsd.edu)

Due Feb 22nd at the beginning of class

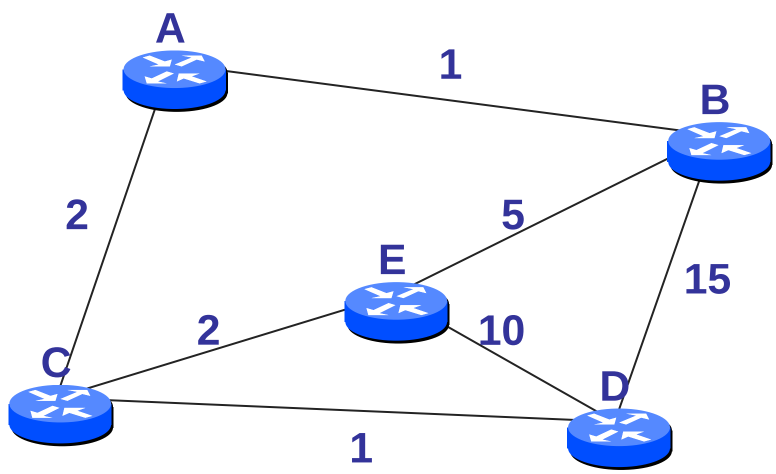
1. Consider the following network. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path from F to all network nodes.



Show how the algorithm works by filling in a table with columns for step, N and D(x),p(x), where step is the iteration through Dijkstra's algorithm, N is the set of nodes whose least-cost path from the source is definitively known, D(x) is the cost of the path from the source to destination x that currently (as of this iteration of the algorithm) has the least cost and p(x) is the previous node (neighbor of x) along the current least-cost path from the source to x. The first two rows in this table are filled out below. Fill out the remaining entries.

step	N	D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(G),p(G)	D(H),p(H)
0		∞	∞	∞	3,F	1,F	6.F	∞
1	E	∞	∞	4,E	2,E		6,F	∞

2. Consider the following network. Assume that each node initially knows the costs to each of its neighbors. Consider the distance vector algorithm and show the distance table entries at node E (this is a table showing the cost to reach each destination via each neighbor).



3. Consider a general topology (**any** potential network) and the distance vector algorithm. Suppose that at each interaction, a node exchanges its minimum costs with its neighbors and receives their minimum costs. Assuming that the algorithm begins with each node knowing on the costs to its immediate neighbors, what is the maximum number of iterations required from this point before the algorithm converges (i.e. that all nodes agree on the shortest path to all destinations)? Justify your answer. *Hint: it may help to think of the network in terms of its diameter d (the maximum number of hops between any two points in the network).*

4. Describe how loops are avoided in the Border Gateway Protocol (BGP)?

5. Consider a router/switch with three input interfaces (A,B,C) and three output interfaces (D,E,F). All interfaces are the same speed, packets are a fixed size and the underlying switch has no speedup. Packets are scheduled from input interface in a rond-robin fashion (first from interface A, then B, etc). Assume that the following packets (labeled by output interface) arrive at each input interface (read from right to left… i.e. EDF means packet F arrives first, then D and so on):
   A: DDF
   B: EFE
   C: FED

Could virtual output queuing help the throughput of this switch? Why or why not? (be precise)

6. Same question as above but the following packets arrive:
   A: FED
   B: EFF
   C: DFE

.