Problems

1. Give two example computer applications for which connection-oriented service is appropriate. Now give two examples for which connectionless service is best.
2. Are there any circumstances when connection-oriented service will (or at least should) deliver packets out of order? Explain.
3. Datagram subnets route each packet as a separate unit, independent of all others. Virtual-circuit subnets do not have to do this, since each data packet follows a predetermined route. Does this observation mean that virtual-circuit subnets do not need the capability to route isolated packets from an arbitrary source to an arbitrary destination? Explain your answer.
4. Give three examples of protocol parameters that might be negotiated when a connection is set up.
5. Consider the following design problem concerning implementation of virtual-circuit service. If virtual circuits are used internal to the subnet, each data packet must have a 3-byte header and each router must tie up 8 bytes of storage for circuit identification. If datagrams are used internally, 15-byte headers are needed but no router table space is required. Transmission capacity costs 1 cent per 10^6 bytes, per hop. Very fast router memory can be purchased for 1 cent per byte and is depreciated over two years, assuming a 40-hour business week. The statistically average session runs for 1000 sec, in which time 200 packets are transmitted. The mean packet requires four hops. Which implementation is cheaper, and by how much?
6. Assuming that all routers and hosts are working properly and that all software in both is free of all errors, is there any chance, however small, that a packet will be delivered to the wrong destination?
7. Consider the network of Figure given below, but ignore the weights on the lines. Suppose that it uses flooding as the routing algorithm. If a packet sent by *A* to *D* has a maximum hop count of 3, list all the routes it will take. Also tell how many hops worth of bandwidth it consumes.

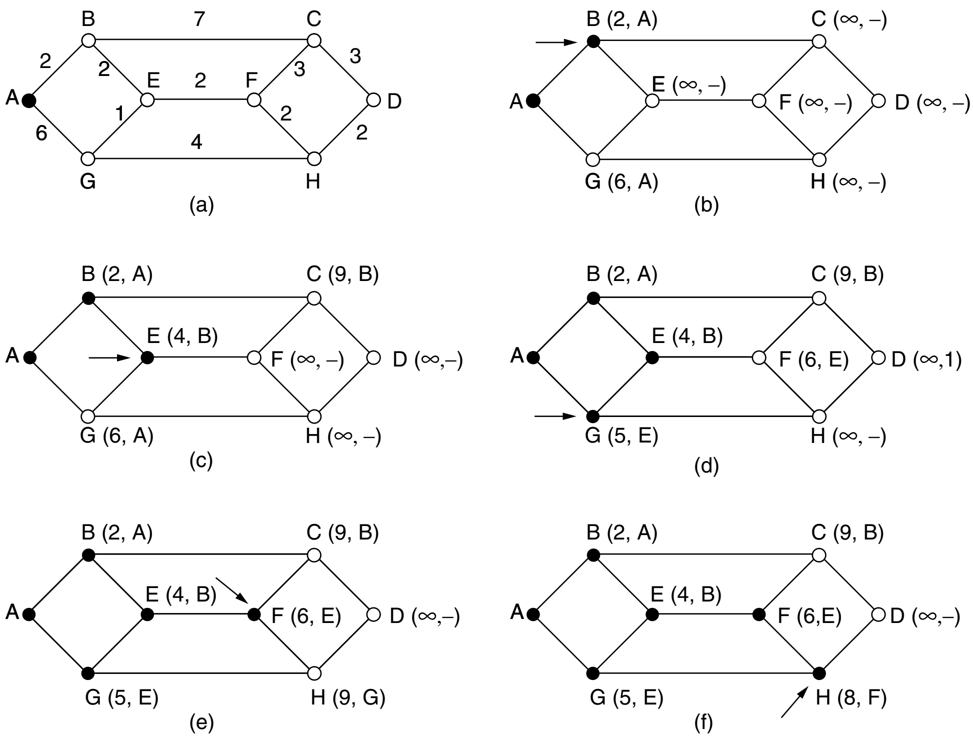

Figure 5-7. The first six steps used in computing the shortest path from *A* to *D*. The arrows indicate the working node.

8. Give a simple heuristic for finding two paths through a network from a given source to a given destination that can survive the loss of any communication line (assuming two such paths exist). The routers are considered reliable enough, so it is not necessary to worry about the possibility of router crashes.