CHAPTER 1 SOLUTIONS

1.1

There are 250,000 pixels per square inch, and multiplying by the number of square inches and the number of bits per pixel gives 5.61×10^8 bits.

1.2

a) There are 16×10^9 bits going into the network per hour. Thus there are 48×10^9 bits per hour traveling through the network, or 13.33 million bits per second. This requires 209 links of 64 kbit/sec. each.

b) Since a telephone conversation requires two people, and 10% of the people are busy on the average, we have 50,000 simultaneous calls on the average, which requires 150,000 links on the average. Both the answer in a) and b) must be multiplied by some factor to provide enough links to avoid congestion (and to provide local access loops to each telephone), but the point of the problem is to illustrate how little data, both in absolute and comparative terms, is required for ordinary data transactions by people.

1.3

There are two possible interpretations of the problem. In the first, packets can be arbitrarily delayed or lost and can also get out of order in the network. In this interpretation, if a packet from A to B is sent at time τ and not received by some later time t, there is no way to tell whether that packet will ever arrive later. Thus if any data packet or protocol packet from A to B is lost, node B can never terminate with the assurance that it will never receive another packet.

In the second interpretation, packets can be arbitrarily delayed or lost, but cannot get out of order. Assume that each node is initially in a communication state, exchanging data packets. Then each node, perhaps at different times, goes into a state or set of states in which it sends protocol packets in an attempt to terminate. Assume that a node can enter the final termination state only on the receipt of one of these protocol packets (since timing information cannot help, since there is no side information, and since any data packet could be followed by another data packet). As in the three army problem, assume any particular ordering in which the two nodes receive protocol packets. The first node to receive a protocol packet cannot go to the final termination state since it has no assurance that any protocol packet will ever be received by the other node, and thus no assurance that the other node will ever terminate. The next protocol packet to be received then finds neither node in the final termination state. Thus again the receiving node cannot terminate without the possibility that the other node will receive no more protocol packets and thus never terminate. The same situation occurs on each received protocol packet, and thus it is impossible to guarantee that both nodes can eventually terminate. This is essentially the same argument as used for the three army problem.