Midterm Exam
CMPSCI 591: Computer Networks
Fall 2000
Prof. Jim Kurose

THIS IS THE SAME EXAM TAKEN BY ON CAMPUS STUDENTS

Instructions:

- **Please use two exam blue books** – answer questions 1, 2 in one book, and the remaining questions in the second blue book.
- Put your name and student number on the exam books NOW!
- The exam is closed book.
- **You have 100 minutes** to complete the exam. **Be a smart exam taker** - if you get stuck on one problem go on to another problem. Also, don't waste your time giving irrelevant (or not requested) details.
- The total number of points for each question is given in parenthesis. There are 100 points total. An approximate amount of time that would be reasonable to spend on each question is also given; if you follow the suggested time guidelines, you should finish with 10 minutes to spare.
- Show all your work. Partial credit is possible for an answer, but only if you show the intermediate steps in obtaining the answer.
- Good luck.

Question 1: ``Quickies'' (24 points, 20 minutes)

- Describe the purpose/functionality of the socket(s) used as the server in reliable, connection-oriented socket-based communication with a client. (You need not discuss client side functionality.)

- What does it mean for a protocol to be *stateful*? What does it mean for a protocol to be *stateless*? Give an example one stateful protocol and one stateless protocol.

- We said that ftp uses *out-of-band control*. What is meant by this?

- What is the difference between a recursive query and an iterated query in the DNS?

- Name five fields that are found in a transport layer segment header, and give a one line summary description of the purpose of each field.

- Suppose a TCP sender receives many successive (i.e., in a row) duplicate acknowledgements for segment *x*. What should the TCP sender infer has happened? Why?

Question 2: Transport Layer Potpourri (24 points, 25 minutes)

- Consider the Alternating Bit (also known as stop-and-wait) protocol. Draw a diagram that shows that if the network connection between the sender and receiver can reorder messages, then the Alternating Bit protocol will not work correctly (make sure you clearly identify the sense it which it will not work correctly). Your diagram should have the sender on the left and the receiver on the right, with the time axis running down the page (as we do in class), showing data (D) and acknowledgement (A) message exchange. Make sure you indicate sequence number associated with any data or acknowledgement segmentd.

- Consider a reliable data transfer that only uses negative acknowledgments. Suppose the sender only sends data infrequently. Would a NAK-only protocol be preferable to a protocol that uses ACKs? Why? Now suppose the sender has a lot of data to send and the end-to-end connection experiences few losses. In this second case, would a NAK-only protocol be preferable to a protocol that uses ACKs? Why?

- What is meant by *network-assisted* congestion control? What is meant by *end-to-end* congestion control? For each case, name a protocol that performs that form of congestion control and briefly (two sentences each) describe the essence of how it works.

- Consider a network in which for each link, the link capacity is greater than the sum of the input rates for all end systems in the network.
  - Is congestion control needed in this scenario? Why?
  - Is flow control needed in this scenario? Why?
  - Would it be better to use circuit-switching or packet-switching in this network? Why?

Question 3: A distributed transaction processing system (26 points, 25 minutes)

Consider a distributed transaction processing system of a client and a remote server. The client receives transaction requests from local users. These transaction request must be communicated to the server, which will satisfy the transaction request and return the result of the transaction request. (You can think of a transaction as requesting an account balance from the server database, and the response containing the account balance. You can think of the transaction as being read-only – that is, executing a transaction will not change the state of the server database*). The client and server communicate over a medium that can lose and delay messages; the maximum delay in the medium is not known.* The medium will *not* corrupt or reorder messages.

The client should receive requests from local users (via the event *callbyuser(request)*) and return results to users (via the event *returndatatouser(data)*) in the order in which the requests were generated. The server receives messages from the client via the *messagefromclient(clientmsg)* event and sends messages to the client via the *messagetoclient(servermsg)* event, where clientmsg and servermsg are messages (that you define) sent from the client and server, respectively.

Give a FSM description of the client and server. Describe the format of the messages sent from client-to-server and from server-to-client. Your protocol should be minimalist in the sense that it should not contain any functionality that is not strictly needed to meet the above requirements.

Question 4: Routing Algorithms (26 points, 20 minutes)

Consider the network shown below.

- Show the operation of Dijkstra's (Link State) algorithm for computing the least cost path from **E** to all destinations. Also, explicitly list the shortest path routes from E to all destinations that are the result of the algorithm's computation.
- Show the distance table that would be computed by the distance vector algorithm in B. Note: you do not have to run the distance vector algorithm; you should be able to compute the table by inspection.