Name: ________________________

CS 325 Computer Networks Fall 2019

Exam #1

Closed book and notes. This exam should have 3 pages.

Problem 1: **[30 Points]** Answer the following questions as concisely as possible.

1a) Why does the Data Link layer have to do framing? Why can't Data Link layer entities just send streams of bits?

1b) Why does the networking library have both a `send` and a `sendto`? When is each appropriate?

1c) What's the difference between IMAP and SMTP? Why do we need both?

Problem 2: You've been hired by Bradco, a soon-to-be-famous software outfit, to help them implement a networked version of the old 20 questions game. They've hired some big names in AI, and now have a server that can randomly select items and then answer questions like "is it bigger than a breadbox?" as players attempt to guess the item. All that's left is to formalize TQTP (the 20 questions transfer protocol) — that's where you come in.

The TQTP application-level protocol should allow clients to transmit questions in English to the server, and receive responses. (It'll run atop a reliable transport-layer protocol, so there are no concerns about data loss.) Bradco anticipates that expert players will become bored after many months of continual play, but market research has shown that players will stay hooked if they are allowed to play shorter games (e.g. they're only allowed 15 questions, etc). Thus, TQTP should allow a player to specify the desired limit when starting a game.

Additional specifics: In some versions of 20 Questions game, the person answering questions is expected to narrow the domain as the game starts (e.g. "It's an animal."). We won't do that. The server won't send any clarifying information at the start of a game. We'll also assume that there's a single "guesser" playing against the server in each game. When the guesser thinks they know the item, they pose their guess as a standard question (e.g. "Is it a moldy cheese sandwich?").

2a) [15 Points]: Describe the message *types* that your TQTP protocol will support and explain why you need them. You don't need to get specific about the format of these message types yet — that's the next part.

2b) [15 Points]: Describe your design for a TQTP packet, including all relevant header details.

2c) [10 Points]: Is your TQTP protocol *stateless*? Explain *briefly* why or why not.

Problem 3: Your boss at Bradco was so impressed with your TQTP protocol design that she asked you to help them design a line of non-standard Ethernet hardware. The design team is trying to finalize their designs, and has asked for your input.

3a) [15 Points]: Their first design expands its contention window *linearly* when collisions occur (adding a constant amount to the window size on each successive collision) but otherwise uses the same mechanism as standard Ethernet. Does a host using this card get more or less than its fair share of the bandwidth if it shares a network with hosts using standard Ethernet? Explain.

3b) [15 Points]: The second design uses the standard Ethernet backoff mechanism, but maintains its contention window value even after a successful transmission. That is, after expanding the window on successive collisions, it stores the window size once it "wins" and uses that stored window size as a starting point again on future transmission attempts. (If we have more collisions we might expand some more, then store the new value.) Does a host using your card get more or less than its fair share of the bandwidth when competing with hosts using standard Ethernet? Explain.