CS270: LAB #20

Constructing final exam question

You may work in teams of ideally three people (two/four is acceptable in the event of an unscheduled absence). Unlike most labs **this lab is due to be submitted into Gradescope at the end of the class period (a10minute leeway is permitted to allow for** upload time etc). In order to receive credit, follow these instructions:

[a] Every team member should be discussing simultaneously the same problem – do NOT try to divvy up the labor and assign different problems to different students since the material is cumulative.

[b] Directly edit this lab PDF with your answers (extra pages can be added in the rare event you need more than the allotted space)

[c] Each lab, rotate which member has the responsibility of being the Scribe. This is the person that is typing the answers and uploading the final PDF – note that only a single copy of the filled in PDF is turned into Gradescope. Only one lab needs to be submitted for the entire team, and all members receive the same score. Make sure to use a font that your PDF editor is compatible with (otherwise you might find your answers appear as weird shapes/sizes or simply disappear entirely!)

[d] The Gradescope submission must have each answer properly tagged with the appropriate question. Moreover, every member of the team must be listed as a submitter. Although it is the Scribe which executes these actions, it is still the responsibility of the entire team to make certain this is done properly (thus it is highly recommended that the Scribe share their screen so the entire team can witness it). Answers which are improperly tagged cannot be seen by the grader and thus cannot be scored.

REMOTE ONLY:

[e] Each lab, rotate which member has the responsibility of being the Recorder. This is the person who hits the Zoom Record button (once the technical permission is granted by the TA/RCF/Professor) and ensures that everyone has their camera/microphone on. They are also the member that is responsible to make sure the DrexelStream video is marked as viewable and entered into the https://tinyurl.com/VidLinkForm webform before 11:59pm (they should also email the rest of their team as confirmation.) Note that the video file doesn't get created/processed until after the Recorder has quit Zoom.

[f] Each lab, rotate which member has the responsibility of being the Manager. This is the person that ensures that everyone is participating equally and honestly, keeps the group on task, ensures that all team members understand a solution before going on to the next question, and presses the "hand up" button in Zoom to summon a TA or the professor (but they only do so after surveying the group to make sure everyone has the same question).

Team Name (CS pioneer): Tim Berners Lee

Scribe name: Cole Bardin

Recorder name: Jeremy Mathews

Manager name: Jackson Masterson

Other team member (if any): Brendan Hoag

#0 [0 pts] Before proceeding further into the lab, have each member take a few minutes to *anonymously* complete the "post course survey" in bbLearn. If we get 75+% participation on this, then every student in the class will receive a bonus point on their final exam. Also, please have already done the four optional questions worth bonus pts on the final.

#1 [10 pts] As a group, collectively decide on a concept that you feel is important in the course but that you found difficult at first (or perhaps still do) and list that concept here:

universal quantifiers

#2 [15 pts] Why do you feel this concept was so challenging? Is there something that could have been done differently in the class to have made it more accessible?

spend more time in class covering this topic, instead of relying on just videos.

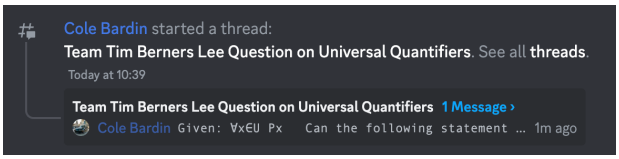
#3 [30 pts] Invent a problem on this topic that you believe would make a fair question to ask on a final exam. It should not simply be a copy of a questions already asked.

Given: ∀x∈U Px

Can the following statement be proven? ∃y∈U ¬Py

#4 [5 pts] Have one of your team members create a thread in the #reviewing Discord channel titled using the template: "Team [CS Pioneer] question on [topic]:... "

Paste a screenshot from Discord of the post below: (Note: do not post the solution yet)



#5 [30 pts] Solve your proposed question:

No it cannot be proven.
If we assume that ∃y∈U ¬Py, we can prove ¬Py.
However, with a universal quantifier elimination, we can prove Py.
Thus a contradiction follows from the assumption, meaning the statement is false.

#6 [1 pt each part] (A) What is a feature/topic in cs270 that you would definitely keep:

Recursion concepts

(B) What is a feature/topic in cs270 that you would want to see changed (how?) /dropped (why?):

Use Scheme or another more widely applicable language. Also drop ProofBuddy, it's way too buggy and unreliable.

Dubnums and graycodes were not really relevant to the rest of the material in the way they were presented.

(C) What is a feature/topic in cs270 that you would want to see added and why:

In class lectures would be significantly more effective. The labs can be shortened and more concise while retaining their effectiveness.

#7 [7 pts] Write a short (1 paragraph) letter to 270 students of next quarter. What advice can you give them to help them succeed? What do wish that someone had told you about this course before you took it? [it is okay if different teammates provide conflicting suggestions -- all points of view matter!]

You may do it on the next blank page, if you wish

Before taking this class, I would take discrete math. It covers the majority of the concepts here in more depth.

I would also be sure to learn about the topics covered in class outside of the context of the course itself. Read up on the practical applications for each of the topics prior to learning them in the course. The course does not do a good job at emphasizing the practicality for each of the topics at hand. Of course recursion is important and has practical applications in engineering and computer science, but the course will not tell you or emphasize why it is important. It very much feels like we're learning in this class just for the sake of learning without understanding the application for whatever concepts we're learning.

With how the course is covered at the moment, you will spend the majority of your time throughout the quarter doing work for this course. The pacing is awful and assumes that this course is the only course you are taking simply by the amount of redundant, monotonous work that has to be done. Start all of the assignments early and be prepared to not have a lot of time left over for other assignments for other classes.

If you must take this class in the way it is currently designed, take it during a term where you can take a fewer number of credits. Reducing your workload as much as possible will reduce all the unnecessary, yet unavoidable stress that this class brings. When it comes to grades, throw away any mindsets of you needing to get Bs or As. To stay sane during this class, your goal should be to pass this class and that is it, otherwise you will be chasing achievements that this class makes impossible despite the claim that it helps you "learn every topic well".

Assuming they keep course corrections the way it is, use it as much as you can, but don't skip classes or stress yourself out about getting to corrections. Chances are, you will have plenty of time after finishing each grade to correct if the delays in grading continue for this class.

Be prepared for Mario Party without bonus stars.