UCLA Computer Science 97 midterm 1 - Spring 2020

Name:_____

Student ID:_____

60 minutes total, 1 minute = 1 point. Open book, open notes, open computer. Answer all questions yourself, without assistance from other students or outsiders.

The exam is not easy, and you are not expected to answer all the questions completely. In your answers, overall approach and intuition will count more than trivial detail. Budget your time while taking the exam. It may help to skip questions that are harder than their point count would suggest.

Print the exam, read the first page, then write your starting time on the first page. Then take at most 60 minutes to answer the questions and write your answers on the exam. (CAE students with x% extra time should add to the 60 minutes accordingly.) When you're done, write your finishing time on the first page, sign the first page, scan the completed exam, and upload your scans to CCLE Gradescope as quickly as you can. If you lack a scanner, carefully photograph the sheets of paper with your cell phone and upload the photographs. Save your filled-out exam until the class is over, and do not give or show it to anybody other than an instructor or TA.

If you lack a printer, read the exam on your laptop's screen, write your answers on blank sheets of paper (preferably 8½″×11″) with one page per question, and upload the scanned sheets of paper. At the end of the exam, you should have scanned and uploaded as many photographs as there are questions. If you do not answer a question, scan a blank sheet of paper as the answer.

The exam is open book, open notes, and open computer. You can use your laptop to use a search engine for answers, and to run programs designed to help you answer questions. However, do not use your computer or any other method to communicate with other students or outsiders, or anything like that. Communicate only via CCLE and Gradescope to obtain your exam and upload your scanned results, or via Zoom or email with the instructor or TAs.

_____ Time (Los Angeles time) you started the exam

_____ Time that you ended the exam

IMPORTANT Before submitting the exam, certify that you have read and abided by the above rules by signing and dating here:

Signature:_____ Date:_____

1 (2 minutes). What is the likely typo in the following shell command, and why is it such a serious typo?

```
rm *.[Oo][Uu][Tt] * .o *.a
```

2. A set of read/write/execute permissions on a file is called “sensible” if the owner has all the permissions of the group, and the group has all the permissions of others. For example, 551 (octal) is sensible, whereas 467 (octal) is not.

2a (2 minutes). Briefly explain why non-sensible permissions don’t make much sense.

2b (6 minutes). How many distinct sensible permissions are there? Explain.

3 (4 minutes). Give two good reasons why backups do not suffice for version control.

4 (4 minutes). Give an example of how renaming a dangling symbolic link can transform it into a non-dangling symbolic link.

5 (8 minutes). Explain how to arrange for Emacs to treat C-t (i.e., control T) as a command that causes Emacs to issue a message like this:

It is now Tue Apr 27 10:30:34 2020.

in the echo area. The message should contain the current date and time.

6 (4 minutes). If a Python class C has a method M and a class variable V, what sort of thing does C.__dict__.items() return and why?

7. By default, the Python expression ‘sys.modules’ signals a NameError, but if you execute ‘import sys’ first the expression does not signal that error.

7a (2 minutes). Briefly explain why not.

7b (4 minutes). Give an example of using the value of the expression ‘sys.modules’ to find out something about your Python process. Your example should explore at least one level past the value of sys.modules itself.

8. Consider the following shell script.

```
#!/bin/sh
atom='[a-zA-Z0-9]+'
string='\\"([^"\]|\\.)*\\"'
word="($atom|$string)"
words="$word(\\.$word)*"
grep -E "$words" | grep ' '
```

8a (4 minutes). Explain briefly what this shell script does, from its user's viewpoint.

8b (3 minutes). How would the script's behavior change if you removed the '-E' from this script?

8c (5 minutes). Modify the original script so that calls grep just once instead of twice, without changing the script's I/O behavior.

9. Consider the following shell transcript:

```
$ git clone https://github.com/git/git.git
Cloning into 'git'...
remote: Enumerating objects: 274, done.
remote: Counting objects: 100% (274/274), done.
remote: Compressing objects: 100% (162/162), done.
remote: Total 286088 (delta 139), reused 215 (delta 112), pack-reused 285814
Receiving objects: 100% (286088/286088), 135.55 MiB | 27.26 MiB/s, done.
Resolving deltas: 100% (212818/212818), done.
```

9a (4 minutes). Say briefly what this command does, and why you might want to run it.

9b (4 minutes). How can it make sense to use Git to get Git's source code, when one needs to start with Git's source code in order to build Git in the first place? Explain why this circular process isn't a fundamental roadblock.

9c (4 minutes). Consider the following continuation of the shell transcript:

```
$ cd git
$ git log | tail -n 5
commit e83c5163316f89bfbde7d9ab23ca2e25604af290
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Thu Apr 7 15:13:13 2005 -0700

    Initial revision of "git", the information manager from hell
```

Git's first version v0.99 wasn't created until July 10, 2005, so how can this Git commit possibly have been created on April 7 of the same year? Briefly explain.