

## UCLA Computer Science 97 Practice Midterm 1 – Spring 2020

100 points total. Open notes, open computer. Forewarning: **this practice midterm is very hard** (like the real one will probably be)!! You are **far** from expected to get perfect on it. Don't stress if you don't know everything – use it as a guide to see what topics you need to spend more time on. Answer what you know first before questions you aren't as sure about.

1. (1 min.) What is **one** main difference between a hard link and a symbolic link?
  
  
  
  
  
  
  
  
  
  
2. (1 min.) Whenever you create a new POSIX directory, it will always have two other special directories inside of it by default. What are they, and what do they do?
  
  
  
  
  
  
  
  
  
  
3. (4 mins.) JavaScript is one of the most popular scripting languages, and is an essential tool for any aspiring web developer. It is an interpreted language; every web browser has its own built-in engine that can understand and execute JavaScript commands line-by-line.

Suppose we have a web browser named Zafari. Its JavaScript engine is simple, and executes JavaScript instructions line-by-line **without** any fancy optimization techniques such as a bytecode compilation step. Compare the execution of JavaScript code in Zafari versus the execution of Python code using the CPython interpreter. Which do you expect to be faster/have better performance?

4. (20 mins.) Internet Protocol version 4 (IPv4) is a core protocol that is used to uniquely address devices with networking capabilities. It performs routing, which is the task of forwarding packets (chunks of data, like text messages, streaming videos, and literally everything else sent over the Internet) from one device to another.

An IPv4 address typically looks like: `[0-255] . [0-255] . [0-255] . [0-255]`, where `[0-255]` is a decimal number from 0 to 255 inclusive. The periods are literal and separate the four numbers. Examples of valid addresses include:

- 127.0.0.1
  - 192.168.0.255
  - 32.45.8.21
- a. (6 min.) Write an extended regular expression that only matches numbers between 0 and 255. It should not match numbers like “02” or “095”.
- b. (2 mins.) Write an extended regular expression that matches IPv4 addresses.
- c. (12 mins.) The shell command `ifconfig`, invoked without any arguments, displays the current configuration for the network interface on your local machine. Write a shell script that looks at the output from `ifconfig`. For each line of output, if it contains any IPv4 addresses your script should output the first (i.e. leftmost) one. Each address you output should be on its own line.
- Hint 1:** Regular expressions are *greedy*; if the address 123.123.123.123 appears, a program like `sed` may match 3.123.123.123. Can you use the context in which addresses appear to fix this?
- Hint 2:** The `-E` option on `sed` may be useful.

5. (10 min.) Consider the shell command being executed on a device named “ubun3” by user “groot”:

```
ubun3:~/zoo-wee-mama groot$ a b < c | ./d > e
```

- a. (2 mins.) What program(s) is/are being executed by the command?
  - b. (3 mins.) For each program you listed above, where will the shell look to find their executables?
  - c. (2 mins.) For each program you listed above, how many arguments is each being invoked with? What is the value of each argument you list?
  - d. (2 mins.) What standard input does each program you listed above receive (if any)?
  - e. (1 min.) Where is the standard output of the above command written to?
6. (25 mins.)
- a. (20 mins.) Write a Python function named `py_spell` whose first argument is a string named `document`, and whose second argument is a **set** of words named `dictionary`. Let us define a *block* as a contiguous sequence of characters separated by spaces, and a *word* as a block with all non-alphabetic characters removed.

Your function should return a **set** of all words in `document` that do not appear in `dictionary`. Words in the dictionary are case-sensitive. If a word in the dictionary contains any non-alphabetic characters, you should remove them before making any comparisons.

For example, if `document =`

```
"Carole Baskin kill-ed her husband!"
```

and `dictionary` contains

```
{ "Carole", "baskin", "killed", "h-u-s-b-a-n-d888" }
```

then your function should output a set containing `{ her, Baskin }`.

- b. (5 mins.) Compare `py_spell` to the `myspell` shell script that you implemented in Homework 1. Given that they are both supposed to be actual spell-checkers, how well does each work, i.e. what words don't work for `py_spell` that do for `myspell`, and vice versa?

7. (14 mins.) Consider the following keystrokes that are typed by a user in the shell in their home directory. In the transcript, a new line means the user typed “Enter”/”return” on the previous line. There is also an “Enter”/”return” after the final line. (Ignore the a, b, c, d, e; those are intended as markers.)

```
echo '#!/bin/bash' > sh
echo 'echo Hello, World!' >> sh (a)
chmod 420 sh
emacs sh (b)
Texttexttext (c)
M-x shell (d)
./sh (e)
```

- a. (2 mins.) What does the file named “sh” contain after the keystrokes on line (a) are typed?
- b. (2 mins.) What happens when the keystrokes on line (b) are typed? Why?
- c. (2 mins.) What happens when the keystrokes on line (c) are typed? Why?
- d. (6 mins.) What happens when the keystrokes on line (d) are typed? In your answer, define what a “process” is and explain exactly what’s happening.

- e. (2 mins.) What happens when the keystrokes on line (e) are typed? Why?
- 
- 8. (10 mins.) Jev Stobs is a nerd who has been tinkering away on his own new operating system named FijiOS. He claims it's extremely fast, but since he's a one-man team he didn't have the time to make his OS POSIX-compliant. What are the benefits and drawbacks of Stobs' approach with FijiOS? If FijiOS takes off, what are the implications for software developers and FijiOS users alike?

9. (5 mins.) Suppose you're working on a project that uses Git for version control. You want to create and switch to a new branch named `feature/localization`, create an empty file named `strings.txt`, then add and commit the new file to this branch, using the commit message "`added strings.txt`". Write a list of shell commands that, when executed in order, will achieve the above.

10. (10 mins.) You're in an alternate universe where Ariana Grande didn't co-write or sing the song NASA. Instead, she became the chief scientist of a highly secretive NASA team. You work under her and help manage terabytes of sensitive data. It is extremely important that this data isn't lost for whatever reason.

Ariana Grande, as a scientist rather than an engineer, doesn't know anything about failure models, and uses a simple policy: Only half of the digital storage available to your NASA team is in use at a time. Every Monday night, the entire contents of those machines is copied exactly as is to the other half of digital storage available. Whatever was in those machines is simply overwritten. Evaluate the effectiveness of Ariana's approach.