CS311 Operating Systems 1

Please read all instructions prior to beginning a given portion of the exam. It is recommended that you read the entire exam prior to starting, as well.

Please write nothing more than your name and ID number on this document. All questions should be answered on the paper provided, with each answer clearly marked. Please put your ID number on the top right of each page that you use. Only a single side of each page will be considered for grading. Please do not make use of both sides.

Part 1

This section will test knowledge of basic operating system principles. These are basic, core concepts that will be important for the remainder of the term.

1. Draw a process tree. This tree consists of 6 processes, named A – F. Process A is the grandparent of process C. Process C is the child of process B, and the parent of process F. Process C can send information via pipes to both process D and process E.

2. There are 2 ways of viewing an operating system, as discussed in class. What are they? Explain one of them.
   
   Extended machine, resource manager.

3. There are multiple methods of IPC. Please name 3 of them.
   
   shared memory, message queues, pipes, sockets, file system, signals

Part 2 - python

This section is all about python. As you likely learned above, important information is contained in the directions. In this section, I will tell you that org is the proper top level domain.

1. What website holds the python documentation?

   python.org

2. Please explain the purpose and usage of the os module. Why is it important?

   It provides operating system independent access to OS functionality
3. What is the purpose and usage of the `sys.argv` list?
   command line arguments

4. There are 2 main ways that python allows interaction with the operating system. Name them.
   subprocess, os

5. The `subprocess` module replaces 5 general methods of spawning or executing new processes.
   Name 3 of them.
   os.system, os.spawn*, os.popen*, popen2.*, commands.*

6. What is list slicing? Please give examples as part of the explanation.
   Extracting specific sections of the list, such as `a[x:y]` pulling item `x` up to but not including `y` out of list `a`

7. Please give a list comprehension that stores the double of all numbers less than 10000 that are themselves perfect squares.
   
   ```
   [2 * x ** 2 for x in range(100)]
   ```

8. Describe the purpose of the `re` module.
   provides regular expression capabilities to python

---

### Part 3 - Regular Expressions

*The questions in this section require that you write a regular expression. As they aren’t going to be used directly in python, don’t worry about escaping, wrapping in strings, etc.*

1. List 6 special characters for regular expressions, and describe what they do. Paired items count singly, for this purpose.
   `$ . * + { } ? [ ] \ ( )$

2. What character class matches any alphanumeric character? What is another way to write it?
   `[a-zA-Z0-9]`, `\w`

3. What is the backslash plague, and how does one get around the issue in python?
   rapid explosion of character escaping, avoid it by using raw strings

4. What is the difference between the `re.match()` and `re.search()` functions in python?
   match only matches at beginning of string, search matches anywhere in string

5. What is grouping, and what does it do for us?
   using () to wrap a sequence, and it stores that portion for later use and retrieval

6. Write a regular expression that will identify valid Roman numerals, using the following information:
   `^M{0,4}(CM|CD|D?C{0,3})(XC|XL|L?X{0,3})(IX|IV|V?I{0,3})$`
   In Roman numerals, there are seven characters that are repeated and combined in various ways to represent numbers.
   - I = 1
• V = 5
• X = 10
• L = 50
• C = 100
• D = 500
• M = 1000

The following are some general rules for constructing Roman numerals:

• Characters are additive. I is 1, II is 2, and III is 3. VI is 6 (literally, 5 and 1), VII is 7, and VIII is 8.

• The tens characters (I, X, C, and M) can be repeated up to three times. At 4, you need to subtract from the next highest fives character. You can’t represent 4 as IIII; instead, it is represented as IV (1 less than 5). The number 40 is written as XL (10 less than 50), 41 as XLI, 42 as XLII, 43 as XLIII, and then 44 as XLIV (10 less than 50, then 1 less than 5).

• Similarly, at 9, you need to subtract from the next highest tens character: 8 is VIII, but 9 is IX (1 less than 10), not VIII (since the I character can not be repeated four times). The number 90 is XC, 900 is CM.

• The fives characters can not be repeated. The number 10 is always represented as X, never as VV. The number 100 is always C, never LL.

• Roman numerals are always written highest to lowest, and read left to right, so the order of characters matters very much. DC is 600; CD is a completely different number (400, 100 less than 500). CI is 101; IC is not even a valid Roman numeral (because you can’t subtract 1 directly from 100; you would need to write it as XCIX, for 10 less than 100, then 1 less than 10).

Part 4 - Extra Credit

1. Using python list comprehensions, write code that calculates the difference between the sum of the squares and the square of the sum for numbers less than 100. Again, this is a single line of python, making use of list comprehensions.

   \[ \text{sum([x for x in range(100)])**2} - \text{sum([x*2 for x in range(100)])} \]

2. This past weekend, a giant in the field has passed away after battling illness. Responsible for many of the concepts that drive modern operating systems, he was an important part of the early days of operating systems, working on both MULTICS and UNIX. Who was he?

   Dennis Ritchie (dmr)