COS 109 Midterm Exam, Fall 2016     In class, Tuesday, October 25

PRINT your name here ____________________________________________

Do not discuss the exam with, or accept help from, anyone. You must write and sign this statement:

“This examination represents my own work in accordance with University regulations.”

Rules

This examination is open-book and open-note:

- You may use the textbook, course notes, your own notes, corrected problem sets and solutions, old exams and answer sheets from the course web page, lab instructions, etc.
- You may use a calculator.
- You may not use anything else; specifically, you may not use a computer, phone or tablet (except that you can use the calculator program on one of these, and you can use your computer to view course notes if you did not print them). You cannot use your computer to access the internet except for the course materials.

Procedure

There are a total of 90 points for the questions; use the point values for each question to allocate your time appropriately (one point per minute).

Write your answers directly on these pages; if you need more space, there is a blank page at the end and you can write on the backs of pages. It's quite all right to be brief as long as you're clear. We have tried to leave plenty of room for answers; if you are writing or computing a lot, you may be off on the wrong track.

Good luck.
1. (15 points) Warmup question

We estimated that there are about 120,000 gas stations in the United States in 2015. We also learned (from Google searching) that there was 1 gas station in the United States in 1913. It is ok to approximate answers to this question. You may want to use the rule-of-72.

(a) What is the annual rate of growth in the number of gas stations over the 102 years from 1913 to 2015 assuming that the rate of growth didn’t change over time

120,000 is about $2^{17}$ and so represents 17 doublings in 102 years. This corresponds to a doubling every 6 years. By the rule of 72, this corresponds to a 12% annual growth rate.

(b) If this rate were to continue into the future, when would we expect there to be about 1,000,000 gas stations in the United States?

1,000,000 is about 8 times 120,000 and so represents 3 doublings. From above, a doubling takes 6 years and so 3 doublings take 18 years. So, at the assumed rate of growth, there would be 1,000,000 gas stations in 18 years, so in 2033.

2. (20 points, 2 each) Short Answers. Write your answer in the space provided.

(a) How many bits are used in the ASCII code for each character and how many characters can be represented in the ASCII character set? Do not distinguish between printable and unprintable characters.

ASCII uses 8 bits and so can represent 256 characters

(b) October 11 is Ada Lovelace day. Why is this relevant to COS 109?

Ada Lovelace was the first computer programmer

(c) For each of the following representations of RGB colors, what are the decimal values for red, green and blue?

\[
\begin{array}{ccc}
\text{ACCEDE} & \text{BEADED} & \text{EFFACE} \\
\hline
\text{AC} & 10*16+12 &= 172; & \text{CE} & 12*16+14 &= 206; & \text{DE} & 13*16+14 &= 222 \\
\text{So, Red} & \text{is } 172, & \text{Green} & \text{is } 206, & \text{Blue} & \text{is } 222 \\
\text{BE} & 11*16+14 &= 190; & \text{AD} & 10*16+13 &= 173; & \text{ED} & 14*16+13 &= 237 \\
\text{So, Red} & \text{is } 190, & \text{Green} & \text{is } 173, & \text{Blue} & \text{is } 237 \\
\text{EF} & 14*16+15 &= 239; & \text{FA} & 15*16+10 &= 250; & \text{CE} & 12*16+15 &= 207 \\
\text{So, Red} & \text{is } 239, & \text{Green} & \text{is } 250, & \text{Blue} & \text{is } 207 \\
\end{array}
\]
(d) Computers today employ the von Neumann architecture. What is the most significant feature of the von Neumann architecture?

In the von Neumann architecture, data and instructions are stored together in computer memory with no distinction between them.

(e) Your laptop computer has a faster CPU and more RAM than the computer that students may have purchased a few years ago. But, it might not have more disk storage. Why is this?

The computer students purchased a few years ago had a mechanical disk (with moving parts). Your disk is solid state. So, the technology has changed.

(f) Give an example of a problem that requires exponential time for its solution.

The Towers of Hanoi problem (moving rings from one post to another with an intermediate post such that a smaller ring is never below a larger ring) requires exponential time.

(g) Give an example of an NP-complete problem.

The knapsack problem where you are given a set of weights and a capacity of a knapsack and asked if the knapsack can be exactly packed is NP-complete.

(h) As the number of transistors in a CPU has grown, caches have been added to the CPU chip. What is the advantage of this?

Information stored in cache is physically closer to the CPU and so can be accessed more quickly leading to faster computation.

(i) The tweet below was sent this past summer. Why do you think WhatsApp settled on “such an oddly specific number”?

256 is the size of a byte and so there is nothing odd about this number.

(j) How many kilobytes in a peta byte?

1 petabyte = 1000 terabytes
1 terabyte = 1000 gigabytes
1 gigabyte = 1000 megabytes
1 megabyte = 1000 kilobytes
So, $10^{12}$ kilobytes in a petabyte.
3. (15 points) Machines

Here is a program in the Toy assembly language, with reminders about what the instructions do.

```
Foo   GET
    get a number from keyboard into accumulator
IFZERO Bar
    if accumulator is zero, go to Bar
LOAD   Sum
    load accumulator with value in location Sum
ADD    1
    add 1 to accumulator
STORE  Sum
    store accumulator in location Sum
GOTO   Foo
    go to instruction labeled Foo
Bar   LOAD   Sum
    print contents of accumulator
PRINT
STOP
Sum   0
    reserve a memory location called Sum, set its initial
    value to 0
```

(a) If this program is given the sequence of inputs 3 -1 4 1 -5 9 2 -7 0 exactly what does it print?

Sum is what is printed; it is calculated by reading an input (until 0 is read), ignoring that number and adding 1 to Sum (which starts at 0).

So, Sum is 8 (the number of inputs read before a 0)

(b) What would happen if the order of the 2 statements ADD 1 and STORE Sum was reversed?

If these two statements were reversed, we would store the value of Sum before adding 1. When we added 1, the value in the accumulator would change. But, we never store this value and so, the result would be that Sum never changed from 0

(c) Starting from the original program, if the command ADD 1 was replaced by the command SUB 1, what would be printed?

If we subtracted one each time through instead of adding 1, our sum would be -8
4. (15 points) State Machines

Design a state machine to sell PopCorn (P) and Candy (C) for tokens(T).

Inputs to the machine are either tokens (T) (all tokens have the same value) or P (asking for popcorn which costs 2 tokens) or C (asking for candy which costs 3 tokens).

Outputs from the machine are P (to signify that popcorn has been delivered) and C (to signify that candy has been delivered).

The machine goes into an Error state if it receives a P or C command and there are not enough tokens to cover the purchase. After a successful purchase, the machine returns to its initial state.

You do not have to worry about giving change. So, e.g. if someone inserts 5 tokens and then asks for popcorn, they get popcorn and you reset to there being no tokens in the machine (as you were at the beginning).

Machine has to count tokens. If there are insufficient tokens and popcorn or candy is requested, machine goes to error state. If there are sufficient tokens, request is delivered and we are done.
5. (25 points, 5 each) Miscellaneous

You need answer only 5 of these 7 questions!! If you do more, please indicate which should be graded.

(a) If you wanted to store the userid (no more than 8 characters), social security number, class year and date of birth (mm/dd/yyyy) all as characters for every undergraduate Princeton student, about how much memory would you need? You can assume that an individual’s information would be stored as a character strings with no effort to simplify. For example the social security number should be taken as a string of 9 characters (rather than as a 9 digit number that could be otherwise simplified). Explain your answer.

Userid is 8 characters, social security number is 9 characters, class year is 4 characters and date of birth is 8 characters. So, 29 characters (or 29 bytes) per student. There are 5300 undergraduates. We can approximate as 5000*30 or 150,000 bytes.

(b) State Moore’s Law. If Moore’s Law continues to hold for the next 12 years, how different will my current machine (which has 16GB of RAM and 1 TB of disk) be at the end of 12 years?

Moore’s Law says that for a fixed cost, the capacity of things (in this case RAM and disk) doubles every 18 months (2 years is also ok here).

In this problem, Moore’s Law would suggest 8 doublings. $2^8 = 256$, so capacity would grow by a factor of 256. RAM would grow to 4096GB or about 4TB; disk would grow to 256TB.

(c) For the sorting algorithm of your choice, describe how it works and give its running time.

In selection sort, we first find the largest, then the second largest, … until we have sorting all the inputs. This requires quadratic $\sim n^2$ operations.

In quicksort, we take the first element and find where it appears in the sorted list by scanning back and forth and have all smaller numbers to one side and all larger numbers to the other side. We then repeat the process on the smaller numbers and on the larger numbers. This requires $\sim n \log n$ operations.

In sort/merge, we divide the list in 2. We sort each half by repeating the process of doing sort/merge on a smaller input until the input becomes so small that it is already sorted. We then merge the two halves by walking down sorted lists. This requires $\sim n \log n$ operations.
(c) For the function \(((\text{NOT } (A \text{ OR } B)) \text{ AND } \text{NOT } C)\)

(i) Draw the circuit (in terms of basic AND, OR and NOT gates) that represents this function

(ii) Give a truth table for the function

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>RESULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

(iii) Find a simpler expression for the function

\((\text{NOT } A) \text{ AND } (\text{NOT } B) \text{ AND } (\text{NOT } C)\)
(d) A car odometer with 6 decimal digits rolls over to zero after 999,999 miles. Suppose that the
odometer in a car works in binary, not decimal.

(i) If the odometer is 12 binary digits long, what binary value does it show just before it rolls
over to zero?

\[ \begin{align*}
111111111111 \\
(ii) \text{ What is that value expressed in decimal?} \\
2^{12} - 1 = 4095 \\
(iii) \text{ What is that value expressed in hexadecimal.} \\
\text{FFF}
\end{align*} \]

(e) Convert the numbers 127, 310, 7835 to hexadecimal, then add them as hexadecimal numbers and
convert the sum to decimal to check your result. DO not use a calculator for this problem and
show all work.

\[ \begin{align*}
127 &= 7*16 + 15(F) = 7F \\
310 &= 1*16^2 + 3*16 + 6 = 136 \\
7835 &= 1*16^3 + 14(E)*16^2 + 9*16 + 11(B) = 1E9B \\
\end{align*} \]

\[ \begin{align*}
7F \\
136 \\
1E9B \\
\hline
2050 \\
2050 \text{ in hex is } 2*16^3 + 5*16 = 2*4096 (8192) + 80 = 8272 \\
127 + 310 + 7835 = 8272
\end{align*} \]

(f) I am looking at 3 algorithms. Algorithm A runs in linear time and performs 100 operations on an
input of size 10. Algorithm B runs in quadratic time and performs 10 operations on an input of
size 10. Algorithm C runs in cubic time and performs 1 operation on an input of size 10.

(i) As the input size grows, will there come a time when algorithm A requires fewer operations
than algorithm B? If so, what is that input size?

On input size 100, algorithm A will require 100*10 = 1000 operations; on input size 100, algorithm
B will require \(10*10^2 = 1000\) operations.

(ii) As the input size grows, will there come a time when algorithm A requires fewer operations
than algorithm C? If so, what is that input size?

On input size 100, algorithm A will require 100*10 operations; on input size 100, algorithm C will
require \(10*10^3 = 1000\) operations.

(iii) As the input size grows, will there come a time when algorithm B requires fewer operations
than algorithm C? If so, what is that input size?

From the above, algorithms B and C will both require 1000 operations on input of size 100.
(h) The picture on the left is a close-up of a seriously geeky t-shirt from Thinkgeek.com. Exactly what does it say?

```
01001000 is 48 in hex, so H
01101100 is 65 in hex, so e
01101100 is 6C in hex, so l
01101100 is 6C in hex, so l
01101111 is 6F in hex, so o
00100000 is 20 in hex, so <space>
01010111 is 57 in hex, so W
01101111 is 6F in hex, so o
01110010 is 72 in hex, so r
01101100 is 6C in hex, so l
01100100 is 64 in hex, so d
00100001 is 21 in hex, so !
```

So,

Hello World!