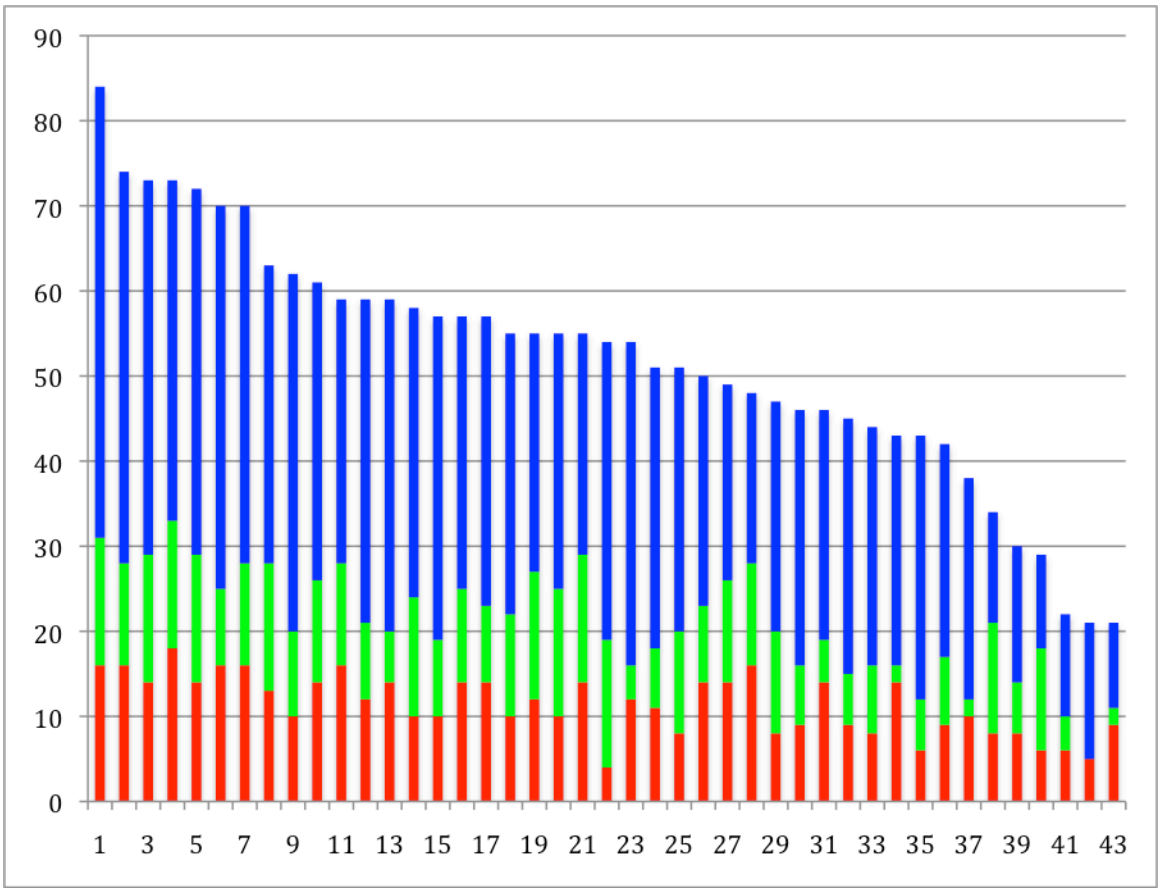
COS 109 Midterm Fall 2018

I graded this myself. The median was 54 and the quartiles were 44 and 59; the corresponding values in 2014 (when I last taught the course) were 57, 44 and 67. As always, the biggest difficulties are with binary numbers, how information is represented in bits, and the like. Algorithm complexity seems a bit shaky. Arithmetic errors were common as well. The three values in each column below are for parts 1, 2 and 3, reading from the bottom.



1. (20 points, 2 each) Short Answers. Circle the right answer or write it in the space provided.

(a) A 7-digit phone number like 5551212 requires 7 bytes if stored as ASCII characters. How many bits are required to store any particular 7-digit number if it is represented as a binary number?

24. The largest possible 7-digit phone number is 9999999, which is greater than 8M (23 bits) and less than 16M (25 bits). Not well handled on average.

(b) In Unicode, the Coptic alphabet goes from **`2C80`** to **`2CFF`**. How many characters are there in Coptic?

128. Ignore the common 2C prefix. FF-80 is perhaps easiest handled by realizing that is 255-128, but includes both ends of the range, so it's really 255-127.

(c) In 1946, John von Neumann said "We are therefore forced to recognize the possibility of constructing a hierarchy of _____, each of which has greater capacity than the preceding but which is less quickly accessible." Which of the following is the proper word to fill in the blank?

accumulators controls instructions memories orders organs processors

memories. Remember the discussion of levels of caches in computers?

(d) The cartoon xkcd.com/1105 shows a geekly criminal with the license plate **`1I1-III1`**. His theory is that no one will be able to correctly record his plate number because of potential confusions between **`1`** and **`I`**. How many such plates are possible?

128. There are 7 positions, each of which is either 1 or I, so 2^7.

(e) If **n** is an integer, how many 1 bits (i.e., bits that have the value 1) are there in the binary representation of the number $2^n \times 2^n$?

1 **2** **n** **2n** **2×2^n** **2^{2n}** **4^{2n}**

1. This is a power of 2, and all powers of 2 have a single 1-bit.

(f) If a 6-inch integrated circuit wafer like the ones passed around in class has 200 chips, about how many of the same chips would there be on a 12-inch wafer?

800. Remember quadratic relationships like area? Double the radius and the area quadruples, and so does the number of chips. Points off for excess precision. Think about the ratio; don't reach for a calculator (program).

(g) On 11/28/05 the NY Times profiled Kazushige Goto [sic], a programmer who hand-crafts mathematical function libraries for scientific computation; his "special skill was in the step-by-step reordering of software instructions to take the greatest advantage of the performance trade-offs offered by each type of [CPU] chip." Which one of these programming tools is Mr. Goto in effect replacing?

assembler **compiler** **debugger** **editor** **interpreter** **simulator**

assembler. He's hand-crafting assembly-language sequences because he can do them better than a compiler can ; note the mention of « each type of CPU chip ».

(h) Suppose a RAM package (like those passed around in class, and analogous to the one pictured here) has 8 chips on each side and each chip contains 2 G bits. What's the total capacity of the package in gigabytes?

4 GB. 8 chips x 2 sides x 2 Gbits / 8 bits/byte.

(i) We used a divide and conquer algorithm to quickly and efficiently count the number of people in Lewis Library 138 one day. Suppose we want to use the same algorithm to count larger crowds. If it takes 10 minutes to count 1,000 people, about how many minutes would it likely take to use the same algorithm to count a crowd of 1,000,000 people? (This is a question about algorithm complexity, not practicality.)

10 **20** **100** **1,000** **2,000** **10,000** **1,000,000** **2,000,000**

20. It's a log n algorithm, as discussed at length in class, and the log of 1M is 20.

(j) Alice says "I've designed a brand new low-power CPU; its instruction set is specifically for cellphones. After I write the assembler for it, no one has to do anything else; people can write C programs for it right away." Bob says "Nonsense. Someone still has to create a C compiler for it." Carol says "You're both wrong. Alice can just use the C compiler that she already uses to compile programs for her Mac." Who's right?

Alice **Bob** **Carol** **none of them**

Bob. No C compiler yet exists for Alice's new CPU and its assembler; Carol's Mac compiler generates code for a Mac, not code for Alice's computer.

2. (15 points) Playing with Toys

Here is a short program in assembly language for the toy computer used in class, with reminders about what the instructions do:

```
FOO   GET          (get a number from keyboard into accumulator)
      IFZERO BAR   (if accumulator value is zero, go to
instruction BAR)
      IFPOS BAZ    (if accumulator value is >= 0, go to
instruction BAZ)
      GOTO FOO     (go to instruction FOO)
BAZ   PRINT        (print value in accumulator)
      IFPOS FOO    (if accumulator value is >= 0, go to
instruction FOO)
BAR   PRINT
      STOP
```

(a) If you run this program and give it the sequence of input numbers **3 1 –4 –7 5 9 –2 –6 8 0**, ***exactly*** what number or numbers does it print, if any?

3 1 5 9 8 0. It's printing the positive number, and the final zero.

(b) Which of these expressions best describes how the running time of the algorithm performed by this program varies as a function of or in proportion to **N**, the number of input numbers?

log N **N** **N log N** **N^2** **2^N** **independent of N**

N. It does the same basic computation on each input number.

(c) The computer described in Von Neumann's 1946 paper includes an instruction that shifts the bits in the accumulator ***to the left*** by one position, replacing the vacated position on the right end by a 0 bit. What arithmetic operation does this shift perform on the binary number in the accumulator?

Multiplies by 2. A fair number of people recognized that it multiplies, but didn't say by what.

(d) If the instruction instead shifts the bits in the accumulator ***to the right*** by one position, discarding the rightmost bit and replacing the vacated position on the left end by a 0 bit, what arithmetic operation does this shift perform on the number in the accumulator?

Divides by 2. Same observation. Some noted that it discards the remainder or rounds down, which is correct, though I did not require it.

3. (55 points, 5 each) Miscellaneous

(a) The familiar multiplication table for decimal numbers has exactly 100 entries.

(i) Write out the analogous multiplication table for binary numbers.

0 x 0 = 0, 0 x 1 = 0, 1 x 0 = 0, 1 x 1 = 1. Not well handled, especially since it's in the book! Those who wrote out the decimal table in binary ignored the warning about writing too much.

(ii) How many entries would there be in an analogous hexadecimal multiplication table? Do not write it out!

256. It's 16 x 16.

(b) The Census Bureau publishes lists of the most popular male and female given names, sorted by frequency so the most popular names are at the top. Suppose that there are **n** names on each list.

(i) Describe an *efficient* algorithm for finding all names, like Alex or Chris, that appear on both lists. Your description should be brief (10-15 words is enough) but very clear about the basic approach or idea.

Sort the combined lists, print names that appear twice. (They will be adjacent.) Or sort one of the lists and use binary search into that from the other list. Each of these is n log n. Some credit for people who gave a clean-enough quadratic algorithm, though the question said "*efficient*", and quadratic algorithms are not efficient when there's a perfectly good n log n algorithm at hand.

(ii) How does the running time of your algorithm depend on **n**, the number of names on each list?

n log n. You have to sort the names; otherwise it's quadratic and that's not efficient.

(c) Name 5 different kinds of transducers that one might find on a modern smartphone.

Vibrator, buttons, screen, accelerometer, microphone, speaker, … Pretty much anything goes here, though not different kinds of buttons or jacks, and not for things that are the same, like "audio recorder" and "microphone".

(d) The display of a really old cell phone represents colors with an RGB model that uses 3 bits for red, 3 bits for green, and 2 for blue; this fits nicely in one byte.

(i) How many different colors can the phone display?

256. There are 8 bits, so 2^8.

(ii) Assuming that the bits are stored left to right in RGB order, what are the hexadecimal representations of **red**, **green**, **blue** and **yellow** (red + green)?

E0, 1C, 03, FC. Much confusion here about what hexadecimal is. Many people wrote an F for each 1 bit and a 0 for each 0 bit; others wrote correct binary but didn't then convert it to hex.

(e) A hard disk like the ones passed around in class spins at a rotational speed of 5,400 revolutions per minute.

(i) Assuming that the number of bits per square centimeter is uniform across the whole disk surface, where should the blocks of a file be located so their contents can be read as rapidly as possible?

near inner edge	**uniformly across whole surface**
near outer edge	**location doesn't affect reading speed**

Near outer edge. More bits pass the read head per revolution because the track is longer.

(ii) If the rotational speed is increased to 7,200 rpm, by what factor does this increase the rate at which bits go past the read heads?

4/3, or 1.33, or 1/3 faster. Most people got this freebie.

(f) A pixel is a picture element and a voxel is a volume element. Suppose you wanted to attach tiny probes all over your body to serve as "touchels", that is, units of touch. (Whether these might be used for sensing or stimulation we will leave to your imagination.) If each touchel is 0.1 inch by 0.1 inch, estimate ***very roughly*** how many touchels there would be on your body. You can use metric units if you prefer; if so, assume that touchels are 1 mm by 1 mm. You must reason quantitatively.

300,000 in inches, 2M in mm? Lots of different body measurements in evidence, but one unifying theme was failing to distinguish between linear and area dimensions. There are 100 touchels in a square inch, not 10; you have to divide by 0.1 x 0.1, for example. Points off for getting that wrong, and for excess precision, which was also common.

(g) The book *The Zero Marginal Cost Society* says that in 2007 there were 10 million sensors connected to the Internet, and that there will be 10 trillion sensors in 2027. (The numbers have been somewhat simplified.)

(i) If this growth is a smooth exponential process, how long does it take for the number of sensors to double?

1 year. It grows by a factor of 10^6 in 20 years, so 10^3 in 10 years, so the doubling time is 1 year.

(ii) What is the rate of growth *per month* of the number of sensors?

6%. If the doubling time is a year, the Rule of 72 implies that it's growing at 6% per month.

(h) Quickies (1 or 2 word answers):

It's possible to write a C compiler in C. True or false?	**True**
I could run Linux in a virtual machine on a Mac. True or false?	**True**
What fast algorithm helped earn its inventor a knighthood and a Turing award?	**Quicksort**
"Bell Labs operating system, 4 letters" (crossword puzzle clue)	**Unix**
A prox card gets its power from radio waves. True or false?	**True**

(i) "One if by land, and two if by sea." Suppose that some modern-day Paul Revere wants to send more extensive information about an invading force. He wants to encode these three items in as few total bits as possible:

- whether the force is coming by land, sea or air
- the approximate size of the force to the nearest 1,000, with a maximum of 15,000
- what time of day or night the force set off, to the nearest hour.

How many total bits does Paul need to use, and why?

11. 2 for land/sea/air, 4 for the 16 possible sizes, and 5 for the 24 distinct times.

(j) The picture on the left shows a pattern of protruding bricks on the wall of a campus building. I've drawn it more clearly on the right. ***Exactly*** what does the pattern say?

P=NP? Most people got this one (which was discussed in the Q/A). It's upper case.

(k) On 12/9/09, the *Wall Street Journal* said that the Nook e-book reader has 2 GB of memory, "enough to hold about 1,500 digital books." On 12/10/09, the *New York Times* said that a zettabyte "is equivalent to 100 billion copies of all the books in the Library of Congress." Assuming that these two statements are correct, compute ***very roughly*** how many books there must be in the Library of Congress.

7,500 or similar, depending on your assumptions. Clearly nonsensical; as one of you noted, it's probably the "failing New York Times" at fault.