Honor Pledge: “I pledge my honor that I have not violated the Honor Code during this examination.”

Please write the pledge in full and sign it:

This examination is open-book and open-notes:
- you may use course materials including the text book, course PowerPoints, your own notes, corrected problem sets and solutions, lab instructions, etc.
- you may use a computer or tablet to access these course materials but not for other purposes.
- you may use a calculator though questions requiring calculation will accept solutions that are approximations as we have done in class.

There are 180 points for the questions; use the point values for each question to allocate your time (one point per minute). There are choices in the second and third sections of the exam. No extra credit will be given for answering extra questions.

If you’re writing or calculating a lot on any question, you may be off on the wrong track.

Write your answers directly on these pages; use the back if necessary. In general, be brief, but if you need more space, attach extra pages and make sure your name is on every extra page. Please write legibly — I can't grade it if I can't read it.

Good luck.
1. **(60 points, 3 each) Short Answers. Write your answer in the space provided.**

(a) What is a pixel? How is the color of a pixel represented? What is a typical resolution of a laptop screen?

A pixel is a picture element which represents color as 3 bytes, one for each of red, green and blue. A laptop screen might be 1280x1024

(b) The Princeton University libraries contain 11 million books. If the books were numbered starting at 1, how many bytes would be required to give each book a unique number?

11 million books would require 24 bits (a thousand is about 10 bits, a million is about 20 bits and 11 fits in 4 bits) and so 3 bytes

(c) What is 419 in binary? In hexadecimal?

419 in hexadecimal is 1A3 and in binary is 110100011

(d) Show the truth table for a 3 input XOR gate

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>X XOR Y XOR Z</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

(e) If a program runs in quadratic time and the input size is doubled, what do you expect will happen to the running time?

Running time is quadrupled (Multiplied by 4)

(f) What is the purpose of functions in a programming language?

Functions are used for operations that are done several times; makes the code easier to follow because series of statements do not have to be repeated

(g) What is middleware?

Middleware is a layer of software that lies between the machine and the user to make the writing of applications easier. APIs are part of middleware

(h) Give an example of an API that you have used.

The prompt() command in Javascript is an API

(i) What is open source software? Give an example of an open source program.

Open source software is software that is freely distributed; firefox is an example of open source software.
(j) What happens when your computer runs out of RAM?

Your computer uses virtual memory which is storage space on disk that behaves like RAM.

(k) What happens when your computer runs out of disk space?

You either need to buy more disk or move files to the cloud.

(l) How many bytes are used to represent a MAC address? How many bytes are used to represent an IPv4 address?

4 bytes for IPv4 and 6 bytes for a MAC address.

(m) What is the difference between a cookie and a super cookie?

A cookie comes from a single application (e.g., your bank saving account number or Amazon holding a shopping cart). A super cookie is shared between applications to transfer information (typically of interest for ad placement) about your habits.

(n) Alan Turing appeared in this course twice. In what contexts did he appear?

The Turing machine (an abstract model of a computer is named after him) and he developed the machine use to break the enigma in World War 2.

(o) Why are there two keys in a public key cryptosystem?

One is the public key which is used to send you a message that is encrypted. The other is the private key that is used by you (and only known by you) to open a message encrypted with your public key.

(p) What are the steps in doing a Google search? You need not give details of how each step is done.

The web is crawled to collect the contents of all web pages focusing on key words on each page; then the keywords on all web pages are indexed so that we can know which pages contain which words; then the search is done to find relevant pages for the search terms; finally the PageRank algorithm is used to determine the importance of pages found.

(q) What is net neutrality?

Net neutrality is the principle whereby everyone’s traffic moves at the same speed over the internet. ISPs cannot provide special access to some customers (e.g., for more money) than for others.

(r) We said that Twitter could disable Donald Trump’s account. Why could they do this?

Trump’s tweets may be viewed as insulting to others and so they violate the Twitter terms of service.

(s) What is the basic unit of storage in a file system? Typically how much is stored in this unit?

The basic unit is the disk block consisting of 1024 bytes.

(t) About how many bananas are there in a pound of bananas?

Dining services says 2.4, we said 4 in class.
2. (80 points) Do 4 of the following 6 problems; each problem counts for 20 points.

(a) The following is the source for a web page with an embedded Javascript program that reads a collection of numbers, tests each to see if it is even and adds together all of the even numbers it sees. This value is then reported by writing its value (eventotal here). The program stops accepting input when the number 0 is input. The % operator computes the remainder when its first operand is divided by its second operand (in this example, num%2 returns 0 if num is even and 1 if num is odd.

Modify the program so that it creates another variable named oddtotal in which it stores the sum of the odd numbers that come as input. Thus, it will compute two values -- eventotal and oddtotal -- where eventotal is (as in the code below), the sum of even numbers input and oddtotal is the sum of odd numbers input. Make sure that your code also writes out both of these values.

```html
<html>
<body>
<P> evens.html: add all even numbers
<p>
<script>
    var eventotal = 0
    var num
    num = prompt("Enter new value, or 0 to end")
    num = parseInt(num)
    while (num != 0)    {
        if (num%2 == 0)
            eventotal = parseInt(eventotal) + parseInt(num)
        num = prompt("Enter new value, or 0 to end");
    }
    document.write(eventotal)
</script>
</body>
</html>
```
(b) For the following program written in the Toy language; study the problem to answer the questions below

```
MORE GET              get an input from the user
STORE SUM1            store the accumulator value as SUM1
IFZERO END            if the accumulator value is 0, go to END
ADD SUM2              add the value SUM2 to the accumulator
STORE SUM             store the accumulator value as SUM
LOAD SUM1             load the value SUM1 into the accumulator
STORE SUM2            store the accumulator value as SUM2
GOTO MORE             go to the instruction labeled MORE
END LOAD SUM          load the value SUM into the accumulator
PRINT                 print the value in the accumulator
STOP                  end program execution
SUM1 0                initialize the value of SUM1 to 0
SUM2 0                initialize the value of SUM2 to 0
SUM  0                initialize the value of SUM  to 0
```

What does the program do? For each of the input sequences

7, 4, 2, 0
6, 12, 1, 1, 0
1, 0

what is the output of the program?

Program reads in a series of numbers (until a zero is read to end the process) and returns the sum of the last two numbers input.

```
7, 4, 2, 0             yields output 6
6, 12, 1, 1, 0         yields output 2
1, 0                   yields output 1
```

(c) For this problem you will deal with integers modulo 3. The value of an integer modulo 3 is the remainder when the number is divided by 3. So, for example, 7 modulo 3 is 1 and 9 modulo 3 is 0. When we count numbers modulo 3, we count as 1, 2, 0, 1, 2, 0, etc. Using this definition of integers modulo 3, we will now design a finite state machine to determine when I have won a certain game of solitaire.

In the solitaire game, I have a deck of cards consisting of kings (K) and queens (Q). The way the game works is that I deal cards from the deck. As I do this, I keep track of the number of kings modulo 3 and the number of queens modulo 3. The first time these two numbers are the same, the game ends and I am declared the winner. The game continues until I have won.

For example, if the cards drawn are KKQKQQ I have won because there were 3 (= 0 modulo 3) kings and 3 queens. Similarly, if the cards drawn are KKQKQKK because 5 (the number of kings) is 2 modulo 3 which equals the number of queens. In each case, this is the first time I will have won.

Your task is to build a state machine that takes as input a string of K’s and Q’s keeping track of the number of kings modulo 3 and the number of queens modulo 3. Before drawing the machine, specify what the states of the machine are, what the input is, ... You can assume that the machine has one final state (which you can label as SUCCESS) and that the machine continues to compute until it reaches this state.
Machine has 8 states: A start state, 6 states labelled (0,1), (0,2), (1,0), (1,2), (2,0), (2,1) which correspond to (K,Q) the current values of the number of kings (modulo 3) and the number of queens (modulo 3) and a final state labelled SUCCESS.

Transitions are that from the start state, a K moves you to state (1,0) and Q to state (0,1). In all other cases, if the value of K is 1 less than the value of Q and a K is input, the transition is to the Success state. Similarly, if the value of Q is 1 less than the value of K and a Q is input, the transition is to the SUCCESS state. Otherwise, transitions when a K is input increase the value of K by 1 and transitions when a Q is read increase the value of Q by 1.

(d) Suppose I put $1 into a savings account that pays 4% interest and $1 into a savings account that pays 6% interest. After 72 years, how much money will be in each account?

By the rule of 72, 72 years would be 4 doublings at 4% interest, since there is a doubling every 72/4 = 18 years and so, that account would have $16. 72 years would be 6 doublings at 6% interest, since there is a doubling every 72/6 = 12 years and so, that account would have $64.

(e) The message below has been encoded by a Caesar cipher of some length. What is the unencoded version of the message?

FRPSXWHU VFLHQFH LV D GHOLJKWIXO VXEMHFW WKDW L HQMRB

COMPUTER SCIENCE IS A DELIGHTFUL SUBJECT THAT I ENJOY
(f) For each of these running times, give an example of an algorithm that runs in that time. Also, explain what happens when the input size grows. You might tell what changes in the running time of the algorithm if the input size doubles or grows by some amount of … or you might tell what has to happen to the input size to make the running time double or grow by a certain amount.

(1) \( \log N \) – binary search (e.g. searching for a number in a sorted list) runs in \( \log N \) time. If the input size doubles, the running time increases by 1.

(2) \( N \) – linear search (e.g. finding all aces in a deck of cards) runs in \( N \) time. If the input size doubles, the running time also doubles.

(3) \( N \log N \) – quicksort (or sort merge which are used to sort a collection of integers run in \( N \log N \) time. If this input doubles, the running time slightly more than doubles.

(4) \( N^2 \) – a simple sort where every number is compared to every other number runs in \( N^2 \) time. If the input size doubles, the running time goes up by a factor of 4.

(5) \( 2^N \) – The Towers of Hanoi problem requires \( 2^N \) time for its solution. In the problem size grows by 1, the time to solve it doubles.
3. (40 points) Do 4 of these 6 questions. Each question counts for 10 points.

(a) Alice and Bob are students want to communicate in a secure fashion avoiding Eve, the evil eavesdropper.

(1) Alice finds Bob’s public key and sends him a message. Without getting into the mathematical details of how this works, describe the idea behind public key cryptosystems and why Alice and Bob can feel comfortable knowing that an eavesdropper will not be able to read the message.

A problem is found where it is hard to solve the problem but easy to check whether a solution is correct. For example, it is difficult to factor a large number but once factors are known, it isn’t hard to confirm that they are the factors of the number. Bob knows the factors of a very large number, Alice only knows the number. From this, two keys are generated, one of which Alice can use to encrypt the message; this is Bob’s public key. To decrypt the message requires Bob’s private key, known only to Bob. So, an eavesdropper can see the message but cannot decode it; it can only be decoded by knowing the solution to a very difficult to solve problem.

(2) Bob receives Alice’s message and wants to be sure that it was actually Alice who sent the message. Explain why he would have this concern and how Alice can sign the message to assure him that she was the one who sent it?

Anyone could have encrypted the message to Bob which only Bob can decrypt. But, if Alice uses her private key to encrypt the message after using Bob’s public key, then the message can be undone by using Alice’s public key (which undoes the effect of her private key) and so leaves a message that only Bob can decrypt. Bob can then decrypt that message with his private key. Only Alice knows her private key and so only Alice could have signed the message.

(b) In this problem, you are asked to supply various details of how the internet works.

(1) To begin, briefly define the following terms

(a) Peering points

Peering points are places where two ISP’s meet to transfer bytes from one network to the other

(b) Root servers

Root servers are servers which are the core of the internet and have complete tables of addresses. Since addresses change slowly, routers along the way will not have their tables of addresses updated as frequently but can refer to root servers when an updated address is needed

(c) TCP/IP

TCP and IP are the protocols that the internet is built on. TCP controls the breaking of a message into packets at the source and the reconstruction of the message from packets at the destination. IP controls the actual routing of the message from source to destination.

(d) ICANN

ICANN is the Internet Corporation for Assigned Names and Numbers which is responsible for the running of the internet, working with domain registries to allow new sites to get IP addresses for their names, controlling the DNS (Domain Name Service, overseeing the root servers, …)

(2) How many root servers are there?

13
(3) Why are we not concerned that terrorists can take down all of the root servers since there are not so many?

Many of the servers are actually anycast which means that they exist in several physical locations. So, in truth, there are many more than 13 root server locations that a terrorist would have to destroy.

(c) Consider the following facts:

(1) It has been estimated that Google has enough bandwidth to move 4 gigabits of information per second

(2) On a recent day, it was estimated that the Google index of the web contained 4.8 billion webpages.

(3) The average webpage contains 2 megabytes of information.

Based on these facts, if Google wanted to do a web crawl to gather the information on all websites once, how long would it take? Note that, in reality, when Google crawls the web, it doesn’t visit each website on each crawl.

Google has to obtain 4.8 billion x 2 million bytes which comes to $4.8 \times 10^9 \times 2 \times 10^6 = 9.6 \times 10^{15}$ bytes which is about $8 \times 10^{16}$ bits (8 bits per byte). 4 gigabits is $4 \times 10^9$ bits per second.

Dividing $8 \times 10^{16}$ bits of information by $4 \times 10^9$ bits per second says that is would take $2 \times 10^7$ seconds. A day is 86,400 seconds, so about $10^5$ seconds which means that a full crawl would take about $2 \times 10^7 / 10^5 = 200$ days.

(d) In this problem, you will show how the memory manager in an operating system works. To begin, suppose I have a machine with 1 GB of RAM. The operating system requires 256 MB to operate. This yields a diagram with 256 MB of RAM in use and the rest free. I have provided you with a picture of this situation at the start. For each of the following actions, modify the diagram on the next page to show what the RAM looks like.

(1) I start Word which begins with 128 MB of RAM and Chrome which requires 256 MB of RAM

(2) My Word document grows and I require an additional 128 MB of RAM

(3) Chrome requires an additional 128 MB of RAM

(4) My Word document grows again and so I require another 128 MB of RAM

(5) I close Chrome

(6) I start Firefox which requires 64 MB of RAM
(e) Part of a file system has the following structure with sizes of files given in bytes:

1. Directory D1 contains directories D2, D3 and D4.
2. D2 contains directory D5 and the file ps1.html (12456 bytes).
3. Directory D5 contains file ps1old.html (12546 bytes).
4. Directory D3 contains files ps2.html (34523 bytes) and ps2ans.html (23424 bytes).
5. Directory D4 contains files ps3.html (42453 bytes) and ps3ans.html (72310 bytes).

Here is a diagram showing the directory tree:

```
  D1
   D2
   D5
|   ps1.html
|   D3
|   ps2.html
|   ps2ans.html
|   D4
|   ps3.html
|   ps3ans.html
```

Each directory requires 1 block of storage in which are recorded the files (including directories) it contains and their sizes.

(i) Determine how many blocks (including directory blocks) are required to store everything under D2:

1 block for D2, 1 for D5, 13 each for ps1.html and ps1old.html, total of 28 blocks.

(ii) How many blocks including directory blocks would the file system have to read to determine if D1/D2/D5/ps1old.html and D1/D2/ps1.html have the same contents?

D1 is 1 block and 28 more as in part (i).

(iii) How many blocks, including directory blocks, would the file system have to read to list the names and sizes of all html files?

5 blocks since only the directories would have to be considered and each is 1 block.
(f) Provide an estimate for the number of cups of coffee consumed in the dining hall of Rockefeller College during reading period. Explain the assumptions that you made in creating your estimate.

There are about 500 students in Rocky and reading period lasts 10 days. If we assume that the average student has 2 cups of coffee a day (some have 4 and some have none), this comes to

$$500 \times 10 \times 2 = 10,000$$

cups of coffee during reading period.