Rules

- Keep at least one empty seat between you and everyone else.
- Please write your answers directly on the pages.
- You may use your notes, your book, a calculator.
- You may **not** discuss the test with anyone else, including online sources (no cell phone, computer).
- You may ask clarification questions during the exam.
1. Sock Sorting Bugs

Each of the sock sorting routines below has a fatal flaw (bug). Match the description of a flaw to the name of the sorting routine that has the flaw.

a. Get two socks, but misplaced the first one before matching.

b. Forgot to actually get socks to match.

c. Put back socks that were already matched.

d. Didn’t put socks back if they didn’t match.

Bad Sock Sorters

```python
def sorterA():
    if not match(a,b):
        replaceSock(a)
        replaceSock(b)

def sorterB():
    a = getSock()
    b = getSock()
    while match(a,b):
        replaceSock(a)
        replaceSock(b)

def sorterC():
    a = getSock()
    a = getSock()
    if not match(a,b):
        replaceSock(a)
        replaceSock(b)

def sorterD():
    a = getSock()
    b = getSock()
    match(a,b)
```
2. Binary Numbers

(A) What is the decimal number 35 in binary?

(B) What is 10010110 in decimal?

(C) What is the two’s complement of 01011001?

3. Exponentially Right?

Which of these sentences use the term “exponentially” correctly?

a. Once I got my cell phone, I was exponentially better at know what time it was.

b. The number of transistors on a chip has increased exponentially over the past twenty years.

c. Home prices in the northeast have been rising exponentially.

d. My new mac is exponentially more powerful than my previous one.
4. Pictures

My Macbook’s screen has three resolutions. The middle one is 1152x720 pixels. How many bits does it take to specify an image on the screen?

5. Logical Construction

Write a logical expression (C = something that can include As, Bs, nots, ands, and ors) to match the truth table.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>False</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>False</td>
<td>True</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>False</td>
<td>True</td>
</tr>
<tr>
<td>True</td>
<td>True</td>
<td>False</td>
</tr>
</tbody>
</table>
6. Random Questions

A. I flip a quarter and a dime together. How many rounds of flipping before both come up heads, on average?

B. I start with the number 10. I flip a coin. If it comes up heads, I subtract one from my number. How many flips, on average, before I reach zero?

C. During the summer season, each day rains with probability $3/7$. How many days go by, on average, before there are 6 rainy days?

D. I have 1 dollar. Each time I play the lottery, there is a $1/100$ chance I double my money. On average, how many times do I have to play before I have $1,024$?

7. Hamming It Up

1. Take the string: yabba-dabba-doo and count up the frequency of each character: $a = ___$, $b = ___$, $d = ___$, $o = ___$, $y = ___$, $- = ___$.

2. Create the Hamming code for these characters: $a = ___$, $b = ___$, $d = ___$, $o = ___$, $y = ___$, $- = ___$.

3. How many bits does it take to write yabba-dabba-doo in your code? ____
8. Song Growth

How many syllables do you sing in \( n \) verses of the following three “songs”? Choose from \( O(n), \ O(n \ lg \ n), \ O(n^2), \) and \( O(n^2 \ lg \ n) \).

Verses are numbered.

a.

b.

c.

8i. A Reduction Story

1. You can build a bit out of silicon.

2. You can build a nand gate out of bits. You can build a bit out of silicon.

3. You can build a logic gate out of nand gates. You can build a nand gate out of bits. You can build a bit out of silicon.

4. You can build a binary adder out of logic gates. You can build a logic gate out of nand gates. You can build a nand gate out of bits. You can build a bit out of silicon.

5. You can build a binary adder out of logic gates. You can build a logic gate out of nand gates. You can build a nand gate out of bits. You can build a bit out of silicon.

6. You can build an expression interpreter out of a binary adder. You can build a binary adder out of logic gates. You can build a logic gate out of nand gates. You can build a nand gate out of bits. You can build a bit out of silicon.
8ii. Mockingbird

1. I’m going to buy you a mockingbird.
2. If the mockingbird won’t fly, I’ll buy you a checkered tie.
3. And if the checkered tie is ripped, I’ll buy you movie script.
4. And if the movie script is dull, I’ll buy you a fishing pole.
5. And if the fishing pole is broke, I’m going to buy you a funny joke.

8iii. Itinerary

1. On day 1, we laughed.
2. On day 2, we cried.
3. On day 3, we spoke.
4. On day 4, we lied.
5. On day 5, we lost.
6. On day 6, we found.
7. On day 7, we freed.
8. On day 8, we bound.
9. Pathology

A. What’s the shortest path from G to F?

B. What’s a longest (cycle-free) path from G to C?

10. Halt or Not

For what initial values of \( n \) does each of these python programs halt?

```python
def hw1(n):
    while n < 10:
        print n
        n = n / 2

def hw2(n):
    while n < 100:
        print n
        n = n + 1

def hw3(n):
    while n > n - 1:
        print n
        n = n - 2

def hw4(n):
    while n < 1024:
        print n
        n = n - 1
```