(a) What is a bot and why were the first bots not only not malicious, but in fact, helpful software?

(b) How can a botnet be used today to launch attacks?
RSA660 is a 660-digit integer which is the product of two primes. It was announced by RSA Security and a $20,000 prize was awarded to the first team who could factor it. That challenge was met in November 2005.

(a) What is the significance of being able to factor large numbers which are the products of two prime numbers in terms of the RSA algorithm specifically?

(b) What is the significance in terms of encryption in general?
Problem: CA8  Area: Security and Reliability

Select any two of the following types of attacks. Then:
   (i) describe how the attack works
   (ii) describe methods to defend against this type of attack.

Man-in-the-middle attack
Meet-in-the-middle attack
Reply attack
Birthday attack

Attack 1:

Attack 2:
Problem: CA9  
Area: Security and Reliability

Select any two of the following potential vulnerabilities. Then:
(i) briefly describe how the vulnerability can be exploited
(ii) methods or techniques to prevent the vulnerability from being exploited

Buffer overflow
Race condition
Pseudo-random number generation

Vulnerability 1:

Vulnerability 2:
(a) What is static code analysis? How is it useful in system security and reliability?

(b) What is the two-terminal problem? How is it important in networked system reliability?

(c) What is reachability? How is it important in networked system security?
For the circuit on the right, assume

PFET: \( W/L = 4/2, \ V_T = -0.4 \ V, \ \gamma = -0.4 \ \text{V}^{1/2}, \ V_{DSAT} = -1 \ \text{V}, \ k' = -300 \ \text{E}-6 \ \text{A/V}^2, \ \lambda = -0.1 \ \text{V}^{-1}; \)

NFET: \( W/L = 4/2, \ V_T = 0.4 \ \text{V}, \ \gamma = 0.4 \ \text{V}^{1/2}, \ V_{DSAT} = 0.6 \ \text{V}, \ k' = 120 \ \text{E}-6 \ \text{A/V}^2, \ \lambda = 0.06 \ \text{V}^{-1}. \)

Some convenient equations (ignoring short channel effects):

NFET:

1. \( V_{GS} < V_T \quad I_{DS} \approx 0 \)
2. \( V_{DS} \leq V_{GS} - V_T \quad I_{DS} \approx k' \frac{W}{L} (V_{GS} - V_T) V_{DS} - \frac{V_{DS}^2}{2} \)
3. \( V_{DS} \geq V_{GS} - V_T \quad I_{DS} \approx k' \frac{W}{L} (V_{GS} - V_T)^2 \)

PFET:

1. \( V_{GS} > V_T \quad I_{DS} \approx 0 \)
2. \( V_{DS} \geq V_{GS} - V_T \quad I_{DS} \approx k' \frac{W}{L} (V_{GS} - V_T) V_{DS} - \frac{V_{DS}^2}{2} \)
3. \( V_{DS} \leq V_{GS} - V_T \quad I_{DS} \approx k' \frac{W}{L} (V_{GS} - V_T)^2 \)

a) \( (33\%) \) What state would you expect the PFET to be in: cut-off, saturation, or linear? Explain.

b) \( (67\%) \) Find the switching threshold, \( V_M \). Specifically, show the state (cut-off, sat., linear) of the

NFET. Hint: \( \sqrt{(a + b)^2} = \pm(a + b), \) also: for \( ax^2 + bx + c = 0, \ x = -b \pm \sqrt{b^2 - 4ac} \ 2a \).
For each of the following questions, circle the appropriate answer and explain your choice.

a) If the width of the transistor increases, the current will:
   - Increase
   - Decrease
   - No change

b) If the length of the transistor increases, the current will:
   - Increase
   - Decrease
   - No change

c) If the supply voltage of a chip increases, the maximum transistor current will:
   - Increase
   - Decrease
   - No change

d) If the width of a transistor increases, the gate capacitance will:
   - Increase
   - Decrease
   - No change

e) If the supply voltage of a chip increases, the gate capacitance of each transistor will:
   - Increase
   - Decrease
   - No change
Design a Recognition Chip that performs the following function:

Step 1) read in an 8-bit \(2^a\) complement number, \(Z\), on port \(X\).

Step 2) read in an 8-bit unsigned number, \(N\), on port \(X\).

Step 3) read in \(N\) consecutive \(2^a\) complement numbers on port \(X\) (i.e. \(B_1, B_2, ..., B_N\)).

Step 4) output a count, \(C\), which is equal to the number of times \(B_i = Z\) or \(-Z\); \(i \in \{1, N\}\)

(if \(N = 0\), no count is output)

Step 5) repeat Steps 1-5

Both the input and output follow the One Cycle Demand Driven handshaking convention. At the rising edge of \(clk\) when \(reset\) is asserted, the chip should reset to its initial state. Show both the datapath and ASM.
Write a VHDL function called SQRT to perform the square root of the std_logic_vector input, Z. Z is in fixed-point fractional format with a range of \([1, 2)\). The output is also in fixed-point fractional format with the same range of \([1, 2)\), and is a std_logic_vector with the same length as Z.

**Square-Root Algorithm:**

\[ S^0 = Z - 1.0 \text{ and } q_0 = q^0 = 1. \]

For each iteration, \( j \in [1, N] \), where \( N \) is the length of \( Z \), if two times the previous partial remainder, \( 2S^{j-1} \), is greater than \( (2q^{j-1} + 2^j) \) then \( q_j = 1 \) and the new partial remainder is: \( S^j = 2S^{j-1} - (2q^{j-1} + 2^j) \), otherwise \( q_j = 0 \) and \( S^j = 2S^{j-1} \). \( q^j = q^{j-1} \) & \( q_j \). In this function we are interested in calculating \( q^{N-1} \). However, we must round \( q^{N-1} \) to make the result more accurate. To do this \( q_N \) is calculated. If \( q_N = 0 \) then the return value is: \( q = q^{N-1} \), otherwise \( q = q^{N-1} + 2^{-(N-1)} \). Using this method, the absolute value of the error (actual square-root – q) is less than \( 2^{-N} \).

**Example:** \( Z = 1.1010_2 = 1.5625_{10} \)

\[
\begin{align*}
\text{j=1} & & 2S^0 = 1.010 & & q_0 = 1 & & q^0 = 1 & & S^0 = 0.1010 \\
\text{j=2} & & 2q^0 + 2^{-1} = 10.1 & & q_1 = 0 & & q^1 = 1.0 & & S^1 = 1.010 \\
\text{j=3} & & 2S^1 = 10.10 & & q_2 = 1 & & q^2 = 1.01 & & S^2 = 0.01 \\
\text{j=4} & & 2q^1 + 2^{-2} = 10.01 & & q_3 = 0 & & q^3 = 1.010 & & S^3 = 0.1 \\
\text{j=5} & & 2S^2 = 1.0 & & q_4 = 0 & & q^4 = 1.0100 & & S^4 = 1.0 \\
\text{j=6} & & 2q^2 + 2^{-3} = 10.1001 & & q_5 = 0 & & & & \\
\end{align*}
\]

result = \( q^4 = 1.01001_2 = 1.25_{10} \rightarrow (\sqrt{1.1010_2} - 1.0100_2 = 0.02475_{10} < 2^{-5}) \)

*note that if \( q_5 \) was 1, then the result would have been \( 1.0101_2 \)}
Write down three examples of speculative execution in modern processors. For each example, clearly state what is speculated, how is the speculative execution detected in the re-order buffer, and how is the execution rolled back if the speculation is incorrect.
Describe the hardware structure of a branch prediction buffer and a branch target buffer. For each structure, draw a diagram to explain the organization of the structure. Also describe how the structure is used to do branch prediction, starting from the time an instruction is fetched and the time it is fully executed.
Consider a 5-stage pipeline consisting of instruction fetch, instruction decode, execute, memory access, and write back stages. Is there any code sequence below that absolutely must be stalled in order to achieve correctness?

Assume that all memory references hit in the cache, and that the processor does all dependency checks. Assume that both forwarding and code reordering are allowed.

Explain your answer. If you use reordering to remove any stalls, please give the reordered code. If you use forwarding to remove any stalls, please indicate what data items are being forwarded. Include the source and destination of any forwarded items.

a)

lw r1, 20(r2)  
sub r3, r1, r2  
add r4, r2, r4

b)

lw r1, 20(r2)  
sub r3, r1, r2  
sw 36(r12), r24  
lw r4, 0(r12)

c)

lw r1, 20(r2)  
sub r3, r1, r2  
sw 36(r12), r24  
lw r4, 0(r2)

d)

add r1, r2, r1  
lw r3, 100(r2)  
sub r5, r6, r7