You do not need to turn in a solution. If you would like feedback on it, please send it to the corrector.

1. Let $f(x, y) = \phi(x - y) + \phi(x + y - 1)$, where ϕ is some "activation function" with one input and one output.
   (a) Draw a circuit for f with $+$, $\times$, and ϕ gates. You may use edge weights for scaling.
   (b) Draw the circuit ∇f obtained by applying Backpropagation to f. The gates in your circuits should be $+$, $\times$, ϕ, and ϕ'. What is the size of this circuit? What is the depth?
   (c) Assume now that ϕ is the function $\phi(t) = \frac{1}{2}t^2$. Then $\phi'(t) = t$ is the identity function. Simplify the circuit by evaluating all gates that depend on constants only (e.g., replace $1 \times (-3)$ by the constant -3) and short-circuiting all ϕ' gates. What is the size and depth of the simplified circuit?
   (d) Explain how the circuit in part (c) computes the function $\nabla f(\mathbf{x}) = A^T(A\mathbf{x} - b)$, where $A\mathbf{x} = b$ is the linear system $x - y = 0, x + y = 1$.
   (e) In general, given a linear system with m equations and n unknowns, provide a reasonably tight upper bound in big-Oh notation on the size, depth, and number of wires of the output of backpropagation given a circuit for $f(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - b\|^2$ as input, and justify it.

2. A secret is 2-out-of-3 shared among Alice (party 1), Bob (party 2), and Charlie (party 3) via Shamir's scheme. The modulus is $q = 5$.
   (a) Bob's share is 4. Charlie's share is 2. What is the secret?
   (b) Alice, Bob, and Charlie want to "rerandomize" their shares without changing the secret. Alice replaces her old share with a random number r modulo 5. Explain how Bob and Charlie should recompute their new shares with Alice's assistance.
   (c) Alice and Charlie want to subtract 1 from the secret without talking to Bob. How should they modify their shares?

3. In this question you will investigate circuits for the millionaires' problem (the argmax function).
   (a) Design arithmetic circuits (with plus and times gates) for the functions NOT x, x_1 AND x_2, and x_1 OR x_2. The gates operate modulo q (for q prime). The circuits should produce the correct output when the inputs x, x_1, x_2 take the values 0 (false) and 1 (true).
   (b) Design a Boolean circuit (with NOT, AND, OR gates) for the less than or equal function
   $$\text{leq}(x, y) = \begin{cases} 1, & \text{if } x \le y, \\ 0, & \text{if } x > y. \end{cases}$$
   Assume the numbers x and y are provided in bit representation as n-bit strings. What is the size of circuit in big-theta notation?
   (c) Use parts (a) and (b) to design an arithmetic circuit for the function
   $$\text{argmax}(x, y, z) = \begin{cases} \texttt{100}, & \text{if } x > y \text{ and } x > z, \\ \texttt{010}, & \text{if } y > x \text{ and } y > z, \\ \texttt{001}, & \text{if } z > x \text{ and } z > y. \end{cases}$$
   Your circuit may output anything in case some two inputs are equal. What is its size in big-theta notation?
   (d) **(Optional)** Implement the BGW algorithm for three parties discussed in class (in arithmetic modulo five). To test correctness, run it (i) on the function $f(x, y, z) = (x + y) \times z \bmod 5$ and (ii) on the circuit from part (d), with x, y, z restricted to 4 bits each. To test security on f, find inputs that satisfy $f(x, y, z) = f(x, y', z')$, sample sufficiently many of Alice's views on these input triplets, and convince yourself that they are identically distributed by comparing their empirical probability mass functions. You are encouraged to use AI assistance for coding.

4. The code $\mathcal{C}$ consists of all 7 bit strings $\mathbf{x} = x_1x_2x_3x_4x_5x_6x_7$ that satisfy these three constraints modulo two:

$$\begin{aligned} x_4 + x_5 + x_6 + x_7 &= 0 \quad \text{and} \\ x_2 + x_3 + x_6 + x_7 &= 0 \quad \text{and} \\ x_1 + x_3 + x_5 + x_7 &= 0. \end{aligned} \tag{1}$$

If the index 1 to 7 is written out in binary representation, each constraint corresponds to a bit of the index, and it involves only those indices for which this bit is set to 1:

$$\begin{aligned} x_{\underline{1}00} + x_{\underline{1}01} + x_{\underline{1}10} + x_{\underline{1}11} &= 0 \\ x_{0\underline{1}0} + x_{0\underline{1}1} + x_{1\underline{1}0} + x_{1\underline{1}1} &= 0 \\ x_{00\underline{1}} + x_{01\underline{1}} + x_{10\underline{1}} + x_{11\underline{1}} &= 0. \end{aligned}$$

(a) Show that for every assignment to $x_3x_5x_6x_7$ there exists a unique assignment to $x_1x_2x_4$ that satisfies all the constraints (1).

(b) Use part (a) to count the number of elements (codewords) in $\mathcal{C}$.

(c) Argue that if $\mathbf{x}$ is in $\mathcal{C}$ and $\mathbf{y}$ differs from $\mathbf{x}$ in position i only then the left-hand sides of (1), when evaluated on $\mathbf{y}$ and read from top to bottom, equal the binary representation of i.

(d) Use part (c) to argue that $\mathcal{C}$ has distance at least 3.

(e) Use part (c) to design an algorithm that corrects up to one error in $\mathcal{C}$: Given $\mathbf{y}$ that differs from a codeword $\mathbf{x}$ in at most one bit it outputs this $\mathbf{x}$.

(f) (**Optional**) Show that up to a permutation of the indices $\mathcal{C}$ is the Hamming $[7, 4, 3]$ code from Lecture 7.