Reminder: You may work in groups of three. Be sure to properly cite any sources you use.

1. **Estimating nonzero error.** In class, we saw that by surveying just 20 members of any size population, we could get \( \approx 89\% \) confidence that a claim was true of at least 90\% of the population. But, this calculation relied on the *absence* of any counterexamples to the claim among the 20 surveyed; indeed, if the claim were actually true of, say, 95\% of the population, then we are somewhat likely to find such a counterexample within a sample of size 20, and this calculation would do us no good. In such a case, we can instead use Chernoff/Hoeffding bounds (Kearns & Vazirani, Theorem 9.2), but this comes at a price. Use the additive form of the bound to determine a sample size such that by counting the fraction of counterexamples to the claim, with 89\% confidence, we obtain an estimate of the true fraction of the population for which the claim holds to within an additive 10\%.

2. **Guessing the size parameter.** Exercise 1.5 in Kearns & Vazirani. (For this problem, it’s helpful to assume the following model of a learning algorithm: instead of the examples being given up-front, the algorithm has an “example oracle,” a subroutine that returns a new labeled example that is independently drawn from the underlying distribution \( D \).)

3. **The most expressive representations.** Recall that we defined a class \( C \) to be (improperly) PAC-learnable by a hypothesis class \( H \) if our PAC-learning algorithm runs in time polynomial in \( n \) (the number of attributes), the size of the target \( c \in C \), \( 1/\epsilon \), and \( 1/\delta \), and with probability \( 1 - \delta \) produces a representation \( h \in H \) as output that agrees with the labels given by \( c \) with probability \( 1 - \epsilon \). We restricted our attention to representations that are efficiently evaluatable, i.e., to classes \( H \) such that there is a polynomial time algorithm that, given as input a representation \( h \) and example \( x \), computes \( h(x) \). Show that if a class \( C \) is improperly PAC-learnable by any efficiently evaluatable hypothesis class, then it is PAC-learnable by Boolean circuits. (**Hint.** You may find it useful that, as stated in lecture, circuits can be efficiently generated for any efficient algorithm.)

4. **Separating the expressive power of DNFs, CNFs, and Decision Trees.** Recall that a DNF is an OR of ANDs of literals (where the ANDs of literals are called “terms”) and a CNF is an AND of ORs of literals (where the ORs of literals are called “clauses”). Consider the following, linear-size DNF formula defining the function \( \text{Tribes}_{2,n/2}(x_1, \ldots, x_n) \)

\[
\text{Tribes}_{2,n/2}(x_1, \ldots, x_n) = (x_1 \land x_2) \lor (x_3 \land x_4) \lor \cdots \lor (x_{n-1} \land x_n)
\]

(a) Describe a CNF formula for \( \text{Tribes}_{2,n/2} \). (**Hint.** It is helpful to consider the extremal cases of inputs where as many of the attributes are true as possible but \( \text{Tribes}_{2,n/2} = 0 \) and where as many of the attributes are false as possible, but \( \text{Tribes}_{2,n/2} = 1 \).)

(b) Prove that any CNF formula for \( \text{Tribes}_{2,n/2} \) requires size \( 2^{\Omega(n)} \). (**Hint.** Consider your solution to 4a. What features of your solution were forced, and why?)

(c) Prove that 4b implies that any Decision Tree for \( \text{Tribes}_{2,n/2} \) must also have size \( 2^{\Omega(n)} \).