1. **PAC Learning Axis-aligned Rectangles in $\mathbb{R}^n$**

   Do Exercise 1.1 of the Kearns-Vazirani textbook. In other words, give an efficient PAC learning algorithm for the class of axis-aligned rectangles in $\mathbb{R}^n$.

2. **Two-Oracle PAC Model**

   Do Exercise 9 in Chapter 3 of the Shalev-Shwartz–Ben-David textbook. In other words, show the equivalence of the standard PAC model and the two-oracle PAC model.

3. **Online to PAC Conversion**

   Recall that the conversion from an online algorithm with mistake bound $m$ to a PAC algorithm given in class works as follows: “Run $A$ on a sequence of examples each drawn independently from $D$: If hypothesis $h$ ever survives $(1/\epsilon) \log \left( \frac{m+1}{\delta} \right)$ consecutive examples without making a mistake, stop and output $h$.”

   Now suppose that you have an online algorithm $A$ with some finite mistake bound $m$, but you don’t know what the value of $m$ is. Explain how you can obtain a PAC algorithm from $A$. What is the best sample complexity (in terms of $m$, $\epsilon$, and $\delta$) that you can achieve for your PAC algorithm?

4. **Variants of Consistent Hypothesis Finder**

   In this problem, we will consider two variants of the notion of a “consistent hypothesis finder” given in class and show that each of them suffices for PAC learning. We will assume throughout this problem that $\mathcal{H}$ is a finite hypothesis class.

   (a) We say that a randomized algorithm $B$ is an “unreliable consistent hypothesis finder” for $\mathcal{C}$ using $\mathcal{H}$ if it has the following performance guarantee: Given any sample of $m$ examples $(x^1, c(x^1)), \ldots, (x^m, c(x^m))$ labeled according to some $c \in \mathcal{C}$, with probability $1/m$ (over $B$’s own internal randomness) $B$ outputs a hypothesis $h \in \mathcal{C}$ that is consistent with all the examples. Prove that an unreliable consistent hypothesis finder can be used to construct a PAC learning algorithm for $\mathcal{C}$.

   (b) We say that an algorithm $B$ is an “almost consistent hypothesis finder” for $\mathcal{C}$ using $\mathcal{H}$ if it has the following performance guarantee: Given any sample of $m$ examples $(x^1, c(x^1)), \ldots, (x^m, c(x^m))$ labeled according to some $c \in \mathcal{C}$ $B$ outputs a hypothesis $h \in \mathcal{C}$ that is incorrect on at most one of the $m$ examples. Prove that an almost consistent hypothesis finder can be used to construct a PAC learning algorithm for $\mathcal{C}$.

5. Let $X$ be the infinite set $\{1, 2, 3, \ldots \}$. Let $P_1, P_2, P_3, \ldots$ be an infinite list of computer programs, each of which takes as input an element $x \in X$ and outputs either 0 or 1. That is, each $P_i$ computes some Boolean function $f_i : X \rightarrow \{0, 1\}$. Assume that the list $P_1, P_2, P_3, \ldots$ can be effectively enumerated, meaning that there is some computer program $M$ which, given a value $i$ as input, outputs program $P_i$. 
Suppose you are learning an unknown function $f$, which is guaranteed to be one of the $f_i$’s, in the online mistake-bound model. Give a learning algorithm which is guaranteed to make $O(\log t)$ prediction mistakes, where $t$ is the smallest index such that $f = f_t$. 