Problem 1: 2-Universal Hash Function Family.

The first couple of problems deal with 2-Universal Hash Function Families.

Define a hash function family as a function \( H \) of the form \( H : X \times X \rightarrow R \), where \( H \) is the set of “hash functions” in the family, \( X \) is the input space and \( R \) the output space of the hash functions. \( H, X, R \) are all finite sets. When the family is understood, \( H(h,x) = y \) is often abbreviated as \( h(x) = y \). Given an input \( x \in X \) we will be interested in hashing it using a random \( h \in H \).

Call a hash function family pairwise independent if for all \( x_1 \neq x_2 \in X \) and \( y_1, y_2 \in R \), \( \Pr_{h \in H} [h(x_1) = y_1 \land h(x_2) = y_2] = \Pr_{h \in H} [h(x_1) = y_1] \Pr_{h \in H} [h(x_2) = y_2] \). Call a hash function family \( 2 \)-universal if for all \( x_1 \neq x_2 \in X \) and \( y_1, y_2 \in R \), \( \Pr_{h \in H} [h(x_1) = y_1 \land h(x_2) = y_2] = \frac{1}{|R|^2} \).

Define maximum collision probability of a hash function family as \( \max_{x_1 \neq x_2 \in X} \Pr_{h \in H} [h(x_1) = h(x_2)] \).

1. Show a trivial example of a uniform hash function family (use \( H = R \)) and a trivial example of a pairwise independent hash function family (use \( X = R \)). Show that a hash function family is uniform and pairwise independent if and only if it is \( 2 \)-universal. Also show that for such a hash function family, the maximum collision probability is \( \frac{1}{|R|^2} \).

2. If \( \mathcal{H} : H \times X \rightarrow R \), is a uniform hash function family what can you say about the size of \( H \), in terms of \( |R| \)? What if \( \mathcal{H} \) is a \( 2 \)-universal hash function family?

3. A function \( f : R \rightarrow R' \) is called regular if for each \( y' \in R' \), \(|\{y : f(y) = y'\}| = |R|/|R'| \). Suppose \( \mathcal{H} : H \times X \rightarrow R \) is a \( 2 \)-universal hash function family and \( f : R \rightarrow R' \) is regular. Show that \( \mathcal{H}' : H \times X \rightarrow R' \), where \( \mathcal{H}'(h,x) = f(\mathcal{H}(h,x)) \) is \( 2 \)-universal. Note that this can be used to shrink the output space of a hash function without affecting the other parameters.

4. A function \( f : X' \rightarrow X \) is called one-to-one if for each \( x \in X \), \(|\{x' : f(x') = x\}| \leq 1 \). Suppose \( \mathcal{H} : H \times X \rightarrow R \) is a \( 2 \)-universal hash function family and \( f : X' \rightarrow X \) is one-to-one. Show that \( \mathcal{H}' : H \times X \rightarrow R' \), where \( \mathcal{H}'(h,x) = \mathcal{H}(h,f(x)) \) is \( 2 \)-universal. Note that this can be used to shrink the input space of a hash function without affecting the other parameters.

Problem 2:

This problem shows why \( 2 \)-universal hash function families are useful for the (public-coin) set lower-bound protocol. (See Lecture 15.)

For \( S \subseteq X \) and \( h : X \rightarrow R \), define \( h(S) \subseteq R \) as \( h(S) = \{h(x) : x \in S\} \). Define \( \text{shrink}(h,S) = |S| - |h(S)| \). Note that \( \text{shrink}(h,S) \geq 0 \). Let \( \text{collision}(h,S) = |\{x_1,x_2 \in S : x_1 < x_2 \text{ and } h(x_1) = h(x_2)\}| \).

1. Show that \( \text{shrink}(h,S) \leq \text{collision}(h,S) \).

2. Suppose \( \mathcal{H} : H \times X \rightarrow R \) has a maximum collision probability \( p \). Show that \( \mathbb{E}_{h \in H}[\text{collision}(h,S)] \leq p|S|^2 \). Using part (1) conclude that \( \mathbb{E}_{h \in H}[\text{shrink}(h,S)] \leq p|S|^2 \).

3. Suppose \( \mathcal{H} : H \times X \rightarrow R \) is a \( 2 \)-universal hash function family, then show that for any \( T \subseteq X \) such that \(|T| = |R|/4 \), \( \mathbb{E}_{h \in H}[\text{shrink}(h,T)] \leq \frac{|R|}{16} \).

4. Use this to argue soundness and completeness of the set lower-bound protocol shown in class. Consider for completeness \( S \subseteq X \) such that \(|S| \geq |R|/4 \) and, for soundness \( S \subseteq X \) such that \(|S| \leq |R|/8 \). (Explain clearly what completeness and soundness mean in this context.)

Problem 3:

Show that \( \mathbb{P} \subseteq \mathbb{P} \). (Hint: Associate a count with the output of a function, such that the count when written in binary is identical to the original output.)
Problem 4:
In this problem you will show that $\sharp P \subseteq FP^{PP}$.

An implicit representation of a binary string $\alpha$ of length $2^m$ is a polynomial sized (in $m$) circuit $A^\alpha$ such that $A^\alpha(i) = \alpha_i$, the $i$-th bit of $\alpha$.

1. Consider a binary string $\alpha$ of length $2^m$. Your task is to count the number of 1s in the string, in polynomial time (in $m$). Show how to do this if you are given an oracle $T_\alpha$, which when given a threshold $\tau$ tells you whether the string has more than $\tau |\alpha|$ 1s.

2. Suppose you are given an oracle $H_\alpha$ which can only answer with respect to the threshold $\tau = \frac{1}{2}$, but allows you to give an implicit description of another string $\beta$ of length $2^m$ and answers whether the string $\alpha \beta$ has more than $\frac{1}{2} |\alpha \beta|$ 1s in it. (That is $H_\alpha(A^\beta) = 1$ iff the string $\alpha \beta$ has more than $\frac{1}{2} |\alpha \beta|$ 1s.) Show how to implement the oracle $T_\alpha$ using access to the oracle $H_\alpha$.

3. Consider the language $L$, such that $L(A^\alpha, A^\beta) = H_\alpha(\beta)$. Show that $L$ is in $PP$.

4. Conclude that given oracle access to the $PP$ language $L$, any function in $\sharp P$ can be computed in polynomial time. i.e., $\sharp P \subseteq FP^L$.

Problem 5 (Extra Credit):
Recall the definition of alternating threshold Turing Machines from class (Lecture 17). Given $M_+ = \text{ATTM}[k, (\exists \geq r, \exists), R]$ (i.e. an ATTM with $k$ alternations between thresholds $\exists \geq r$ and $\exists$, and a relation $R$ at the leaves; the degrees of the different $\exists \geq r$ ans $\exists$ configuration nodes are left out of the notation for clarity), with $r > \frac{1}{2}$, define it’s complementary ATTM $M_- = \text{ATTM}[k, (\exists \geq r, \forall), \overline{R}]$. Such a pair $(M_+, M_-)$ is said to decide a language $L$ if $x \in L \iff M_+(x) = 1, M_-(x) = 0$ and $x \notin L \iff M_+(x) = 0, M_-(x) = 1$.

Also recall the definition of an $\text{AM}[k]$ protocol defined by a verification procedure for Arthur, $A$ (and the lengths of the $k$ messages, alternating between random strings from Arthur and messages from Merlin, starting with one from Arthur). an AM protocol $A$ is said to decide a language $L$ with error probability at most $\epsilon$ if $x \in L \iff \max_M \Pr[A \text{accepts } x \text{ after interacting with } M] \geq 1 - \epsilon$ and $x \notin L \iff \max_M \Pr[A \text{ accepts } x \text{ after interacting with } M] \leq \epsilon$.

1. Given an $\text{AM}[k]$ protocol $A$, define a pair of complementary ATTM$s$ $(M_+, M_-)$ as $M_+ = \text{ATTM}[k, (\exists \geq 3, \exists), R]$ and $M_- = \text{ATTM}[k, (\exists \geq 3, \forall), \overline{R}]$, (with degrees of the configuration nodes being the message lengths of the protocol to the power of 2) with $R = A$ and $r = \frac{3}{4}$. Show that if $A$ is an AM protocol that decides a language $L$ with error probability at most $2^{-(k+3)}$, then $(M_+, M_-)$ decides $L$.

Hint: First try $k = 2$. Consider the protocol’s tree, and define the maximum-average acceptance probability for each node (as shown in class). For $x \in L$, using completeness guarantee, what can you say about the fraction of first messages that lead to a node with acceptance probability greater than $1 - 4\epsilon$? For $x \notin L$ use soundness guarantee.

2. Given a pair of complementary ATTM$s$ $(M_+, M_-) = (\text{ATTM}[k, (\exists \geq 3, \exists), R], \text{ATTM}[k, (\exists \geq 3, \forall), \overline{R}])$, (with degrees of the configuration nodes being powers of 2) define an $\text{AM}[k]$ protocol with $A = R$ (and lengths of the messages being logarithms (base 2) of the degrees of the ATTM pair). Show that if $(M_+, M_-)$ decides a language $L$ and if $r \geq 1 - \frac{1}{16}$, then $A_R$ is an AM protocol that decides $L$ with error probability at most $1/4$.

Hint: For $x \in L$, using $M_+$, what can you say about the maximum-acceptance probability of nodes of the constructed protocol’s tree. First try $k = 2$. To extend to general $k$, consider two levels at a time, and use the “union-bound” inequality $(1 - p)^t \geq 1 - pt$. 

CS 579: Computational Complexity
Homework 5
Page 2 of 2