Computer Science 345
The Efficient Universe

Homework 8
Due Wednesday, April 26, 2006

**You may collaborate with other students,
but you should write up the solutions entirely on your own.**

1 One-way functions

Definition 1 *A function* $f : \{0,1\}^* \longrightarrow \{0,1\}^*$ *is a one-way function if*

1. *It is easy to compute* f*: There exists a polynomial time algorithm computing* f*.*
2. *It is hard to invert* f*. Namely, for every polynomial* $p(\cdot)$ *and a probabilistic polynomial time algorithm (adversary)* A*, and for every* sufficiently large *number* n *(i.e. there exists* N *such that for every* $n \geq N$*)*

$$\Pr_{x \in \{0,1\}^n} \left(A(f(x)) \in f^{-1}[f(x)] \right) < 1/p(n).$$

Remark:

$$f^{-1}[f(x)] = \{y : f(y) = f(x)\}.$$

Definition 2 *A function* $f : \{0,1\}^* \longrightarrow \{0,1\}^*$ *is a one-way permutation if the following conditions hold.*

1. *It is easy to compute* f*: There exists a polynomial time algorithm computing* f*.*
2. *It is hard to invert* f*. Namely, for every polynomial* $p(\cdot)$ *and a probabilistic polynomial time adversary* A*, and for every* sufficiently large *number* n

$$\Pr_{x \in \{0,1\}^n} \left(A(f(x)) = x \right) < 1/p(n).$$

3. *The function* f *is a length preserving bijection: (i)* f *is a bijection; (ii) for every* n *the image of* $\{0,1\}^n$ *is* $\{0,1\}^n$*.*

Problem 1 Prove that every one-way permutation is a one-way function.

Problem 2

A. Prove or disprove. For every one-way functions f and g:

1. $h(x) = f(x)\#g(x)$ is a one-way function.

Remark: # denotes concatenation e.g. "0101"#"11" = "010111").

2. $h(x, y) = f(x)\#g(y)$ is a one-way function.

3. $h(x) = f(g(x))$ is a one-way function.

B. Prove or disprove. For every one-way permutation f and every length-preserving bijection g computable in polynomial time:

1. $h(x, y) = f(x)\#g(y)$ is a one-way permutation (x and y have the same length).

2. $h(x) = f(g(x))$ is a one-way permutation.

3. $h(x) = g(f(x))$ is a one-way permutation.

2 $\mathcal{BPP}$

Problem 3 Let L be a language in $\mathcal{BPP}$; and let $A(x, r)$ be a probabilistic algorithm deciding whether "$x \in L$" with error probability $< 1/3$ (here r is the random input). Suppose that our random generator is "corrupted". Namely, it produces biased bits $\tilde{r}_i$: $\tilde{r}_i = 1$ with probability $11/20$ ($\tilde{r}_i$ are mutually independent). We are interested if we still can use A to determine whether "$x \in L$". Is it true that, if $x \in L$, then

$$\Pr\left(A(x, \tilde{r}) = 1\right) > 1/2 \ \ ?$$

Suggest another algorithm B, which can use A, that will decide L using the biased coins $\tilde{r}_i$ with error probability $< 1/3$.

3 Circuits

Problem 4 Define the majority function $\mathrm{Maj} : \{0,1\}^* \to \{0,1\}$ as follows:

$$\mathrm{Maj}\left(x_1, \ldots, x_n\right) = \begin{cases} 1 & , \text{ if } x_1 + \cdots + x_n > n/2; \\ 0 & , \text{ otherwise.} \end{cases}$$

1. Construct a circuit of size $O(n)$ computing the majority function.

2. Prove that the size of any circuit computing Maj is at least $\Omega(n)$; and the depth is at least $\Omega(\log n)$.