COS 433 — Cryptography — Homework 4.

Boaz Barak

Total of 120 points. Due October 25th, 2007.

Exercise 1 (20 points)**.** Complete the proof of Yao's Theorem given in class. That is, prove that a distribution X over $\{0,1\}^m$ is pseudorandom if and only if it is *unpredictable*, where the latter means that for every $i \in [m]$, poly-time A and poly-bounded ϵ,

$$\Pr_{x \leftarrow_{\mathrm{R}} X}[A(x_1,\ldots,x_{i-1}) = x_i] \leq 1/2 + \epsilon(n)$$

You can use the following lemma that we proved in class:

Lemma 1. *Let H^i denote the distribution whose first i bits come from X and the rest $m-i$ bits are chosen uniformly and independently at random. If there is a T-time algorithm D such that*

$$\left|\Pr[D(H^i)=1] - \Pr[D(H^{i-1})=1]\right| \geq \epsilon$$

then, there is a $T+n$-time algorithm P such that

$$\Pr_{x \leftarrow_R X}[P(x_1,\ldots,x_{i-1}) = x_i] \geq 1/2 + \epsilon$$

Exercise 2 (30 points)**.** Complete the proof of the Goldreich-Levin Theorem given in class. That is, prove that if f is a one-way permutation then the following algorithm G is a pseudorandom generator: $G(x,r) = f(x) \| r \| \langle x,r \rangle$ (where $\|$ denotes concatenation and $\langle x,r \rangle \overset{\text{def}}{=} \sum_i x_i r_i \pmod 2$). You can use the following lemma that we proved in class:

Lemma 2. *There is a poly$(n,1/\epsilon)$-time algorithm that given oracle access to an oracle A that computes the function $r \mapsto \langle x,r \rangle$ with probability $1/2+\epsilon$ over the choice of r, outputs x with probability at least $\left(\frac{\epsilon}{100n}\right)^2$.*

See footnote for hint[1]

Note: In both of these questions, the proof is by reduction. A proof by reduction generally has two parts:

1. The description of the reduction— an algorithm B that uses a hypothetical algorithm A as a black box. The description should include the input, operation, and output of B, where the operation clearly states what are the inputs that B feeds into A. You should also explain why B will run in polynomial-time if A does.
2. Analysis of the reduction— proving that if A breaks the security of some crypto primitive X then B breaks the security of another crypto primitive Y.

Make sure you write both parts clearly and precisely.

[1] **Hint:** First argue that there is no way an algorithm P can predict any one of the first $2n$ bits of G's output. Then, using standard Markov-like reasonings, argue that if P succeeds in predicting the $2n+1^{st}$ bit from the previous $2n$ bits with probability at least $1/2+\epsilon$, then with probability at least $\epsilon/2$ over the choice of x_0, P will still have prediction success at least $1/2+\epsilon/2$ conditioned on $x = x_0$. Call strings x_0 with the latter property "good" and use Lemma 2 to argue that you can transform P into an algorithm that inverts the one-way permutation f on good inputs.

Review of group and number theory

The following questions are meant to prepare you for Tuesday's lecture. These questions are self contained, so you can solve them without reading outside sources. Nevertheless, I recommend you also take a look at some of the following resources: **(1)** Katz-Lindell book, Chapter 7 and Appendix B, **(2)** Victor Shoup's book (available online, see course's webpage): pages 1–10 and pages 180–184 are particularly relevant, but look also in Chapter 2 up to and including Section 2.5, first 2 pages of Chapter 7, Chapter 10 up to and including 10.4.1, first two pages of Chapter 11, Chapter 12 and Chapter 13 **(3)** The mathematical background appendix of my upcoming book with Sanjeev Arora also contains some basic number theory background— see link on the course's website.

A *group* $(S, \star)$ is a set S with a binary operation $\star$ defined on S for which the following properties hold:

1. **Closure**: For all $a, b \in S$ it holds that $a \star b \in S$.

2. **Identity**: There is an element $e \in S$ such that $e \star a = a \star e = a$ for all $a \in S$.

3. **Associativity**: $(a \star b) \star c = a \star (b \star c)$ for all $a, b, c \in S$.

4. **Inverses**: For each $a \in S$ there exists an element $b \in S$ such that $a \star b = b \star a = e$.

The *order* of a group, denoted by $|S|$, is the number of elements in S. If the order of a group is a finite number, the group is said to be a *finite group.* If a group $(S, \star)$ satisfies the *commutative law* $a \star b = b \star a$ for all $a, b \in S$ then it is called an *Abelian group.*

Exercise 3 (10 points)**.** Let $+_n$ denote addition modulo n (e.g., $5 +_3 6 = (5 + 6) \mod 3 = 2$). Let $Z_n = \{0, 1, 2, \ldots, n-1\}$. Prove that $(Z_n, +_n)$ is a finite Abelian group for every natural number n.

Exercise 4 (10 points)**.** Prove that for every group:

1. The identity element e in the group is **unique**.

2. Every element a has a **single** inverse.

Exercise 5 (10 points)**.** Let a be an element in a group and let a^{-1} denote the (unique) inverse of a. Then, for every natural number k we define:

$$a^k \overset{\text{def}}{=} \begin{cases} \underbrace{a \star a \star \ldots \star a}_{k}, & \text{if } k > 0; \\ e, & \text{if } k = 0; \\ (a^{-1})^{-k}, & \text{if } k < 0. \end{cases}$$

Prove that for any integers m, n (not necessarily positive) it holds that:

1. $a^m \star a^n = a^{m+n}$.

2. $(a^m)^n = a^{nm}$.

Exercise 6 (10 points)**.** Let $(S, \star)$ be a group and let $S' \subseteq S$. If $(S', \star)$ is also a group, then $(S', \star)$ is called a *subgroup* of $(S, \star)$. Prove that:

1. If $(S, \star)$ is a finite group and $a \in S$ then there exists $m \geq 1$ such that $a^m = a^{-1}$.

2. If $(S, \star)$ is a finite group and S' is a subset of S such that $a \star b \in S'$ for ever $a, b \in S'$, then $(S', \star)$ is a subgroup of $(S, \star)$.

Exercise 7 (15 points)**.** Let a and b be two positive integers. We denote by $\gcd(a, b)$ the greatest common divisor of a and b; i.e, $d = \gcd(a, b)$ if d is the greatest integer that divides both a and b. The extended Euclidean algorithm computes the gcd as follows:

```
input: $a > b > 0$
$r_{-1} \leftarrow a$
$r_0 \leftarrow b$
for $i = 1, 2, \ldots$ till $r_i = 0$
    $r_i \leftarrow r_{i-2} \mod r_{i-1}$
output $r_{i-1}$
```

1. Prove that this algorithm indeed outputs the gcd of a and b.
2. Prove that if d is the gcd of a and b, then there exist (not necessarily positive) integers x, y such that $d = xa + yb$. Can you compute these numbers?
3. For 5 extra points run the above algorithm to compute $\gcd(a, 456)$ where a is your university ID number. Write all the intermediate values of the r_i's.

Exercise 8 (10 points)**.** A group S is *cyclic* if there exists an element $g \in S$ that "generates" the group; that is, $S = \langle g \rangle$, where $\langle g \rangle \overset{\text{def}}{=} \{g^k : k \geq 1\}$. (Such an element is referred to as a *generator* of the group.)

1. Give an example of a cyclic group.
2. Give an example of a finite group that is not cyclic.

In both cases you should prove that the given group is indeed cyclic (resp. non-cyclic).