Name _____ Student No. _____

No aids allowed. Answer all questions on test paper. Use backs of sheets if necessary.

Total Marks: **60**

[20] 1. A shift in paradigms often accompanies the design of security applications. Explain the three fundamental shifts in paradigm.

Solution: An *algorithms* centered approach becomes a *protocol* centered approach. Protocols still use algorithms, but they express an interaction between several entities. *Absolute correctness* becomes *correctness in terms of probabilities*. For example, the Rabin-Miller algorithm for primality returns the right answer with probability at least $1/2$ when p is a composite. *Computational difficulty* becomes an ally rather than an adversary; it can be used to construct difficult to break cryptographic protocols.

[20] 2. One of the challenges in implementing PKCs is computing large powers of an integer modulo a number p. For example, we may need to compute $a^n \pmod p$ where a, b, p are numbers consisting of $\sim 1,000$ bits. This can be accomplished feasibly with *repeated squaring*; explain how this works.

Solution: The first thing to notice is that integers are given in binary representation, that is, the number n is given as $(n)_b = c_r \dots c_1 c_0$ where $c_i \in \{0, 1\}$. For example, $(5)_b = 101$ because $(1 \cdot 2^2) + (0 \cdot 2^1) + (1 \cdot 2^0) = 5$. Now we repeatedly square a as follows:

$$\begin{aligned} a_0 &= a \pmod p \\ a_1 &= a_0^2 \pmod p \\ a_2 &= a_1^2 \pmod p \\ &\vdots \\ a_r &= a_{r-1}^2 \pmod p \end{aligned}$$

where $r + 1$ is the length of the binary encoding of n, i.e., $r + 1 = |(n)_b|$. Notice, and this is very important!, that all the intermediate steps give numbers in $[p]$, that is, the size of the intermediate integers is always bounded by p.

Now $a^n \pmod p$ can be computed as follows:

$x \leftarrow 1$
for $i = 0$ to $i = r$ do
    $x \leftarrow x \cdot a_i^{c_i} \pmod p$
end for return x

[20] 3. Suppose that $f : \mathbb{N} \times \mathbb{N} \longrightarrow \mathbb{N}$ is a function with the following properties:

- for all $a, b, g \in \mathbb{N}$, $f(g, ab) = f(f(g, a), b) = f(f(g, b), a)$
- for any g, $h_g(c) = f(g, c)$ is a one-way function

Explain how f could be used for public key crypto in the style of Diffie-Hellman.

Solution: Suppose that we have a one-way function as in the question. First Alice and Bob agree on a public g and exchange it (the eavesdropper knows g therefore).

Then, let Alice generate a secret a and let Bob generate a secret b. Alice sends $f(g, a)$ to Bob and Bob sends $f(g, b)$ to Alice. Notice that because h_g is one-way, an eavesdropper cannot get a or b from $h_g(a) = f(g, a)$ and $h_g(b) = f(g, b)$.

Finally, Alice computes $f(f(g, b), a)$ and Bob computes $f(f(g, a), b)$, and by the properties of the function both are equal to $f(g, ab)$ which is their secret shared key. The eavesdropper cannot compute $f(g, ab)$ feasibly.