**Future Projects after #6**

**1) (a)** Implement Diffie-Hellman key exchange using the finite cyclic group Gp, where p is a large safe prime, is generated by

g = A = 1 1

1 0.

This is a group of 2 by 2 matrices. The product AB is computed modulo p.

**(b)** Use the matrix A to compute some Fibonacci numbers mod p, using modular exponentiation.

**Remarks**

Det(A) = -1, so Det(An) = (-1)n = ±1, which gives the group very nice properties. For example, it is its own inverse.

**2) (a)** General case g = A = P -Q

1 0,

and Det(A) = Q, so Det(An) = Qn 🡪±∞, not so nice!

We of course reduce modulo p, so that

Det((An)%p) = Qn %p, so |Det((An)%p)| < p.

**(b)** Explore the connection with the so-called Lucas sequences, which generalize the Fibonacci numbers.

Un = PUn-1 – QUn-2

Un = P -Q Un-1

Un-1 1 0 Un-2

Un = An-1 U1

Un-1  U0

Use the matrix A to find some Us and Vs mod p.

Recall: (U0 , U1)= (0, 1), and (V0, V1) = (2, P).

**Use the modular exponentiation method to compute Ak (mod p).**

**See the end of lecture note #1.**

For the sake of comparison, compute the Vs using the Lucas chain (scanned in from book) and compute the Us from the Vs by Um = Δ-1(2Vm+1 – PVm).

Also compute the Fibonacci numbers 0, 1, 2, 3, 5, …from the Lucas numbers

2, 1, 3, 4, 7, 11,… which are the Vs—that is (V0, V1) = (2, 1). (P, Q) = (1, -1) in both cases.

We want to apply the primality test to p. Let’s choose P and Q so that the Jacobi symbol (D/p) = -1. The Jacobi symbol may be computed in a time similar to Euclid’s algorithm—see the Wikipedia article “Jacobi symbol.”

Here D is the discriminant of the equation x2 – Px + Q, which is (P2 – 4Q)/2.

Then the test is Up+1 ≡ 0 (mod p).

**Theorem**

If G is a finite cyclic group (G, x), then G is isomorphic to the additive group (G’, +) = Z|G|, the integers modulo n = |G|.

This produces the following paradox:

\* The multiplicative group (G, x) can be used to produce a secure Diffie Hellman key exchange, if for example, n = |G| is a large safe prime.

\*The additive group (G’, +) will produce a decidedly insecure key exchange.

Just go through the protocol. A discrete log algorithm is not needed. Euclid’s algorithm suffices. (Easy exercise for the reader.)

Question: How can this paradox be resolved?

**3)** Implement Diffie-Hellman key exchange using the additive group of Zp, where p is a prime. In other words, Multiplication in G is achieved by addition in (Zp, +).

Thus if g, h ϵ G, then gh = g + h in (Zp, +).

For example, if p = 7, then G = {0, 1, 2, , 4, 5, 6}, and G’s multiplication table is:

**Multiplication Table for G when p = 7**

g \h 0 1 2 3 4 5 6

0 | 0 1 2 3 4 5 6

1 | 1 2 3 4 5 6 0

2 | 2 3 4 5 6 0 1

3 | 3 4 5 6 0 1 2

4 | 4 5 6 0 1 2 3

5 | 5 6 0 1 2 3 4

6 | 6 0 1 2 3 4 5

Discuss the security of a Diffie-Hellman key exchange protocol based on this G when p is a 500-bit safe prime.