ECE Information and Network Security
Homework 1
Spring 2006

Chapter 2:

Chapter 2: 9, 23

1. Among the shifts of EVIRE, there are two words: arena and river. Therefore, Anthony cannot determine where to meet Caesar.

2. The inverse of 9 (mod 26) is 3. Therefore, the decryption function is $x = 3(y-2) = 3y-2$ (mod 26). Now simply decrypt letter by letter as follows. U = 20 so decrypt U by calculating 3 * 20 - 6 (mod 26) = 2, and so on. The decrypted message is cat.

5. Let $mx+n$ be one affine function and $ax+b$ be another. Applying the first then the second yields the function $a(mx+n)+b = (am)x+(an+b)$, which is an affine function. Therefore, successively encrypting with two affine functions is the same as encrypting with a single affine function. There is therefore no advantage of doing double encryption in this case. (Technical point: Since $\gcd(a, 26) = 1$ and $\gcd(m, 26) = 1$, it follows that $\gcd(am, 26) = 1$, so the affine function we obtained is still of the required form.)

9. If $x_1 = x_2 + (26/d)$, then $\alpha x_1 + \beta = \alpha x_2 + \beta + (\alpha/d)26$. Since $d = gcd(\alpha, 26)$ divides α, the number $\alpha/26$ is an integer. Therefore $(\alpha/d) * 26$ is a multiple of 26, which means that $\alpha x_1 + \beta = \alpha x_2 + \beta$ (mod 26). Therefore x_1 and x_2 encrypt to the same ciphertext, so unique decryption is impossible.

23. The number of seconds in 120 years is

$$60 \times 60 \times 24 \times 365 \times 120 \approx 3.8 \times 10^9.$$

Therefore you need to count $10^{100}/(3.8 \times 10^9) \approx 2.6 \times 10^{90}$ numbers per second!

Chapter 2 Computer Problems

2. Use 'fr=frequency(lcll);' to get a frequency count. Observe that the most common common letter is l, which is 7 places after e. Try shifting back by 7 using 'shift(lcll,-7)' to get the answer 'ans = eveexpectseggsforbreakfast'.

6. a) The message can be found in the file ciphertexts.m under the variable gaat. The following code performs the conversion to 0, 1, 2, 3, and performs the shifting.

```
ind0=find(gaat==A);
ind1=find(gaat==C);
ind2=find(gaat==G);
ind3=find(gaat==T);
vec=gaat; vec(ind0)=0; vec(ind1)=1; vec(ind2)=2; vec(ind3)=3;
vec=mod(vec+1,4);
ind0=find(vec==0);
ind1=find(vec==1);
ind2=find(vec==2);
ind3=find(vec==3);
output=vec;
output(ind0)=A;
output(ind1)=C;
output(ind2)=G;
output(ind3)=T;
output=char(output);
```

The answer is TCCAAGTGTTGGTGCCAACCGGGAGCGACCCTTTCAGAGACTCCGA.

b) The following code assumes that the affine cipher is of the form y = ax+b (mod 4). The parameters a and b should be entered in. The restrictions are that a is relatively prime to 4 which means that a is either 1 or 3.

```
ind0=find(gaat==A); ind1=find(gaat==C); ind2=find(gaat==G); ind3=find(gaat==T);
vec=gaat;
vec(ind0)=0; vec(ind1)=1; vec(ind2)=2; vec(ind3)=3;
vec=mod(a*vec+b,4);
ind0=find(vec==0);
ind1=find(vec==1);
ind2=find(vec==2);
ind3=find(vec==3);
output=vec;
output(ind0)=A;
output(ind1)=C;
output(ind2)=G;
output(ind3)=T;
output=char(output);
```

Chapter 3

1. (a) Apply the Euclidean algorithm to 17 and 101:

$$101 = 5 \cdot 17 + 16$$

$$17 = 1 \cdot 16 + 1.$$

Working back yields $1 = 17 - 16 = 17 - (101 - 5 \cdot 17) = (-1) \cdot 101 + 6 \cdot 17$.

(b) Since $-101 + 6 \cdot 17 = 1$, we have $6 \cdot 17 \equiv 1 \pmod{101}$. Therefore $17^{-1} \equiv 6 \pmod{101}$.

5. (a) $4883 = 1 \cdot 4369 + 514 \cdot 4369 = 8 \cdot 514 + 257 \cdot 514 = 2 \cdot 257 + 0$. Therefore, the gcd is 257. (b) We know that both numbers have 257 as a factor. This yields $4883 = 257 \cdot 19$ and $4369 = 257 \cdot 17$.

6. (a) The first two steps of the Euclidean algorithm are

$$F_n = 1 \cdot F_{n-1} + F_{n-2}$$

$$F_{n-1} = 1 \cdot F_{n-2} + F_{n-3}.$$

It continues in this way until

$$2 = 2 \cdot 1 + 1$$

$$1 = 1 \cdot 1 + 0.$$

Therefore, the gcd is 1.

(b)

$$11111111 = 1000 \cdot 11111 + 111$$

$$11111 = 100 \cdot 111 + 11$$

$$111 = 10 \cdot 11 + 1$$

Thus $11 = 11 \cdot 1 + 0$. Hence gcd is 1.

(c) The first step of the Euclidean algorithm is

$$a = 10^{F_{n-2}} \cdot b + c,$$

where c consists of F_{n-2} repeated 1s. Continuing in this way, in each step we divide F_{j-1} repeated 1s into F_j repeated 1s and get a remainder consisting of F_{j-2} repeated 1s. Eventually, we get down to the computations of part (b), and then obtain that the gcd is 1.

Supplemental

1. Notice that if $n = 7 \cdot 11$ then the numbers a that are relatively prime to n are precisely those that do not contain either 7 or 11 as their factors. The numbers that have either 7 or 11 as factors are those that are multiples of 7 or multiples of 11. Thus, the numbers a with $gcd(a, n) = 1$ are

$$1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 13, 15, \cdots.$$

If we generalize to $n = pq$ where p and q are primes, then we must remove all multiples of p or q. Hence, we remove

$$p, 2p, 3p, \cdots$$

and

$$q, 2q, 3q, \cdots.$$