CS70, Spring 2002
Midterm 2
Vazirani

Problem 1. (Each 5 pts.) Short questions

(a) Find gcd(3n+2, 2n + 1), where n is a positive integer.

(b) For your RSA public key must use $N = p \cdot q = 55$, and an exponent e which is either 5 or 9. Which of 5 or 9 should be your public exponent? Why?

(c) Suppose you use a polynomial $p(x)$ of degree 5 over the field GF_{13} as your error-correcting code (to encode the message $p(0)$). You happen to know that you correctly received the values $p(1), p(2), p(3), p(10), p(11)$, but all the remaining values were corrupted during transmission. Can you recover the message $p(0)$? How?

(d) Find a pair of integers x and y such that $21x+54y = 1$, or prove that no such pair exists.

(e) Compute $2^{110001111100001}$ (mod 23).

Problem 2. (25 pts.)

RSA Joe Hacker decides that he wants to have two public-private key pairs to be used with RSA - that way if one of his private keys is compromised, half his communication is still secure. For convenience, he decides to use a common composite
n = pq (p,q primes) and selects two separate encryption exponents e_1 and e_2, giving two distinct decryption keys d_1 and d_2. He makes e_1, e_2, and n public. You can assume that e_1 and e_2 are relatively prime.

Suppose two people send the same secret plaintext message m to Joe, one encoded by e_1 and the other with e_2. Explain how an evil eavesdropper can efficiently determine m if he intercepts both these messages. What is the running time of your algorithm?

(Hint: use the fact that e_1 and e_2 are relatively prime).

Problem 3. (25 pts.) Secret Sharing

(a) (5 points) Five people wish to share a secret $0 <= s <= p - 1$ where p is a prime, such that any two of them can reconstruct the secret, but no one person has any information about s. Briefly sketch a scheme for achieving this (i.e. what is each person's share of information, and how do two of them recover the secret).

(b) (10 points) Suppose that Alice, Bob and Charlie wish to share a secret $0 <= t <= p - 1$, such that no two of them have any information about t, but all three of them can perfectly reconstruct it. Here is a scheme to do this: Alice's share is a random number r_1 (mod p), Bob's share is a random number r_2 (mod p) and Charlie's share is $r_1 + r_2 + t$ (mod p). Give a method by which Alice, Bob and Charlie can together reconstruct t. Prove that any two of them have no information about t.

(c) (10 points) Now suppose you wish to share the original secret s among seven people: Alice, Bob and

Charlie, and four others. The conditions are that Alice, Bob and Charlie together with any one of the other four can reconstruct the secret; also any two of the other four (other than Alice, Bob and Charlie) can also reconstruct the secret. But any smaller combination (say Alice and Bob and one of the other four) should have no idea about the secret. Give a scheme for achieving this. Why does it work?

Problem 4. (15 pts.) Decidability

You receive a piece of code *C* in your email, but you are not sure whether it was sent as a prank. So before running it, you want to test whether it will send a mail message from your account. Can you write a program Test, such that Test(C) = BAD if C sends a mail message when executed, and OK otherwise. Justify your answer.