CS 70 First Midterm 10 July 2012

NAME (1 pt): _____

SID (1 pt): _____

TA (1 pt): _____

Name of Neighbor to your left (1 pt): _____

Name of Neighbor to your right (1 pt): _____

Instructions: This is a closed book, closed calculator, closed computer, closed network, open brain exam, but you are permitted a 1 page, double-sided set of notes, large enough to read without a magnifying glass.

You get one point each for filling in the 5 lines at the top of this page.

Write all your answers on this exam. If you need scratch paper, ask for it, write your name on each sheet, and attach it when you turn it in (we have a stapler).

1	
2	
3	
4	
5	
6	
Total	

1

Question 1 (20 points) Potpourri.

1.1 (14 points). True or False

For each of the following propositions, circle either T if it is always true, F if it is always false. You do not have to justify your answer. We let $\mathbb{N} = \{0, 1, 2, ...\}$ denote the non-negative integers, $\mathbb{Z}+ = \{1, 2, 3, ...\}$ denote the positive integers.

T F $(p \wedge \neg q) \wedge (\neg p \vee q) \wedge r$

Answer: *False*

We can see this is false by looking at the truth table:

p	q	r	$p \wedge \neg q \wedge (\neg p \vee q) \wedge r$
T	T	T	F
T	T	F	F
T	F	T	F
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	F
F	F	F	F

T F $\forall x, y \in \mathbb{N}, z \in \mathbb{Z} + .[2x \equiv 2y \ (mod\ z) \implies x \equiv y \ (mod\ z)]$.

Answer: *False*

Counterexample, let $x = 3, y = 7, z = 8$. Then, $2(3) \equiv 2(7) \ (mod\ 8)$. However $3 \not\equiv 7 \ (mod\ 8)$

T F Given d pairs $(x_1, y_1), ..., (x_d, y_d)$, with all the x_i distinct, there is a unique polynomial $p(x)$ of degree d such that $p(x_i) = y_i$ for $1 \leq i \leq d$

Answer: *False*

There is a unique polynomial of degree $d - 1$. There exists infinitely many polynomial function of degree d.

T F $(\forall x \in \mathbb{N})(\exists y \in \mathbb{N})(x^6 = y^2)$

Answer: *True*

Simply take $y = x^3$.

T F In a stable marriage instance, if a man and women have each other last on their respective lists, they are guaranteed not to be paired in any stable pairing.

Answer: *False*

T F $\forall x, y \in \mathbb{N}$, if $9x \equiv y \pmod{26}$, then $x \equiv 3y \pmod{26}$.

Answer: *True*

T F $\forall x, y \in \mathbb{N}$, $\gcd(2x, 3y) = \gcd(x, y)$.

Answer: *False*

Consider $x = 3$, $y = 2$.

1.2 (6 points). Prove or disprove the following proposition:
If every even number greater than 2 is the sum of 2 prime numbers, then every odd numbers greater than 7 is the sum of 3 prime numbers.

Answer: This proposition is true. Consider some odd number, o, greater than 7. We can express o as the sum of 3 and some even number e that is greater than 4, $o = 3 + e$. If we can express e as the sum of 2 primes, p and q then $o = 3 + p + q$. (Note: 1 is not considered a prime number).

Question 2 (20 points) Induction: Poor Valentine.
A school teacher is preparing for Valentine's Day. She has instructed each student to make a Valentine. Then, to ensure fairness, she will have the everyone play the following game:
The students all wander randomly around the room for ten minutes, then gives his or her Valentine to the student closest to them. To further ensure fairness, the teacher adds the following rule: each student must keep every other student a different distance away from them.

Show that if the teacher has an odd number of students, one of them will not receive a Valentine. You may assume the class has more than one student.

2.1 (3 points). State and prove the base case for the induction.

Answer: Let $P(n) =$ "for a class of n students, one will not receive a Valentine at the end of the game." We will prove by induction on n, the number of students, that, if there are an odd number of students, one will not receive a Valentine. Formally, we will show that $\forall n \in \mathbb{N}.(n \text{ is odd } \implies P(n))$.
Base Case: $n = 3$.
Let s_1, s_2, s_3 be the students. After wandering around for ten minutes, WLOG, assume s_2 is the closest student to s_1. Because we assume that each student is a different distance away from s_1, we know that s_2 is unique. This means that s_1 will give their Valentine to s_2. Since distance is symmetric, s_1 will also be the closest student to s_2, and thus will give their Valentine to s_1. This means that s_3 cannot receive a Valentine, since all possible Valentine's he or she could have received have already been given to another student.

2.2 (3 points). State the induction hypothesis.

Answer: Suppose for some odd $k \in \mathbb{N}$, $P(k)$.

2.3 (14 points). Complete the proof by stating and proving the induction step.

Answer: Consider a class of $k+2$ students, $s_1, s_2, \ldots, s_{k+2}$. Let s_x be the student closest to s_1 at the end of the game. This means that s_1 and s_x will exchange Valentines. Using our IH, we know that out of the remaining k students, one will not receive a Valentine.

Therefore, for a class of n students, if n is odd, one student will not receive a Valentine. □

Question 3 (20 points) Stable Marriage.
For each of the following questions,

i) Fill in the tables with an example preference list, give a stable pairing for $n = 3$, and show that your example is correct.

ii) Generalize your example to an arbitrary n and show that your generalization is correct. You do not need to provide the stable pairing for your generalization.

3.1 (10 points). Consider a stable marriage instance where *Traditional Propose & Reject Algorithm* returns a pairing which is optimal for both male and female AND terminates after exactly 1 day.

Answer:

Man	highest→lowest		
1	B	A	C
2	C	B	A
3	A	C	B

Men's preference list

Woman	highest→lowest		
A	3	2	1
B	1	3	2
C	2	1	3

Women's preference list

At the first day, there are no conflicts so algorithm terminates with all three couples. Running the *Traditional Propose & Reject Algorithm* where all men proposes, we get $(1, B), (2, C), (3, A)$. Likewise, when female proposes, we get the same pairing $(B, 1), (C, 2), (A, 3)$

In general, with n men and n women, we set up the preference list such that for all $1 \leq i \leq n$, m_i has w_i first on his preference list and w_i has m_i first on her preference list. Then, if we run 1 iteration of stable marriage every m_i will propose to w_i, therefore, every woman will get exactly one proposal. The algorithm will terminate after that step and generate a pairing, T. We know that $\forall i \in [1, 2, ...n].(m_i, w_i) \in T$ which tells us that this pairing is both male and female optimal (they both got their first choice).

3.2 (10 points). Consider a stable marriage instance where *Traditional Propose & Reject Algorithm* returns a "female optimal" pairing AND terminates after exactly n days.

Answer:

Man	highest→lowest		
1	A	B	C
2	A	B	C
3	A	C	B

Men's preference list

Woman	highest→lowest		
A	1	2	3
B	1	2	3
C	1	2	3

Women's preference list

Let's look at the pairing output by the algorithm, then convince ourselves that it is female optimal.

Days	Women	Proposals
1	A	**1**,2,3
	B	-
	C	-
2	A	**1**
	B	**2**,3
	C	-
3	A	**1**
	B	**2**
	C	**3**

This will output the pairing $(1, A), (2, B), (3, C)$. We see that it is female optimal as follows: we know woman A is with her optimal partner because 1 is her first choice. Now, consider a pairing where woman B is paired with man 1, who she likes better than her spouse. In this case $(1, A)$ will become a rogue couple so the couple $(1, B)$ cannot exist in a stable pairing. So we know the female optimal pairing has the couples $(1, A)$ and $(2, B)$. Therefore, we can say the complete female optimal pairing is $(1, A), (2, B), (3, C)$.

In the case where the women are proposing, this works as long as all the women have the same preference list. In every iteration i, $n - i + 1$ women will propose to the same man, m_i. One will get a maybe and the rejected $n - i$ women will cross m_i off their lists and propose to m_{i+1} the next day.

Note, we can use a similar generalization even if the men are proposing. Consider the case where every w_i has the preference list, $\{m_1, m_2, ..., m_n\}$ and every m_i has the preference list $\{w_1, w_2, ..., w_n\}$. Running the *Traditional Propose & Reject Algorithm* still takes n iterations and returns a pairing T where, $\forall i \in [1, 2, ...n].(m_i, w_i) \in T$. But how are we sure it is female optimal? Running the *Traditional Propose & Reject Algorithm* when the women propose will result in the exact same pairing and is guaranteed to output a female optimal pairing. Therefore, this preference list is both male and female optimal and running the *Traditional Propose & Reject Algorithm* will take n days regardless of who is doing the proposing.

Question 4 (15 points) Modular Arithmetic.

4.1 (10 points). Use the extended gcd algorithm to find $\gcd(47, 20)$ and find the numbers x and y where $47 * x + 20 * y = \gcd(47, 20)$

Answer:

e-gcd(47,20)
calls e-gcd(20,7)
calls e-gcd(7,6)
calls e-gcd(6,1)
calls e-gcd(1,0)
returns 1= 1*0+0*0
returns 1 = 6*0 + 1*1
returns 1 = 7*1 + 6*(-1)
returns 1 = 20*(-1) + 7*3
returns 1 = 47*3 + 20*(-7)

4.2 (5 points). Find a value for x that solves $47x \equiv 8 \pmod{20}$. Show your work.

Answer: From 4.1 we see that $47^{-1} \equiv 3 \pmod{20}$. Multiplying both sides by 47^{-1} we get

$$\begin{aligned} x &\equiv 47^{-1} * 8 \pmod{20} \\ &\equiv 3 * 8 \pmod{20} \\ &\equiv 24 \pmod{20} \\ &\equiv 4 \pmod{20} \end{aligned}$$

Question 5 (15 points) RSA.

5.1 (5 points). Alice wants to send Bob a message $m = 5$ using his public key ($n = 26, e = 11$). What cipher text $E(m)$ will Alice send?

Answer:

$$\begin{aligned}
5^1 &\equiv 5 \bmod 26 \\
5^2 &\equiv 25 \bmod 26 \\
&\equiv -1 \bmod 26 \\
5^4 &\equiv (-1)^2 \bmod 26 \\
&\equiv 1 \bmod 26 \\
5^8 &\equiv 1 \bmod 26 \\
5^{11} &\equiv 5^8 \cdot 5^2 \cdot 5^1 \bmod 26 \\
&\equiv 1 \cdot -1 \cdot 5 \bmod 26 \\
&\equiv -5 \bmod 26 \\
&\equiv 21 \bmod 26
\end{aligned}$$

5.2 (10 points). Is Two Always Better than One?

One day, Joe Hacker decides that he wants to improve the security of RSA. He uses $N = pq$ as usual, but has each person send a message with a *different* exponent, e.

Suppose Alice and Bob each send the same message encrypted with their respective public keys, (N,e_1) and (N,e_2), and that Eve intercepts both encrypted messages, $c_1 = m^{e_1} \pmod N$ and $c_2 = m^{e_2} \pmod N$.

Show how Eve can use c_1, c_2, and both public keys to recover the original message m if e_1 and e_2 are relatively prime. (*Hint:* Consider using egcd)

Answer: Run the extended GCD on e_1, e_2, obtaining coefficients r, s such that $re_1 + se_2 = \gcd(e_1, e_2)$. The RHS is one, since the keys are relatively prime. So if we compute:

$$[m^{e_1}]^r \times [m^{e_2}]^s = m^{e_1 r + e_2 s} = m^1$$

Question 6 (20 points) Error Correcting Codes.

6.1 (5 points). Evaluate $f(x) = 3x^2 - x + 1$ on every point of *GF(5)*.

Answer:

x	f(x)
0	1
1	3
2	1
3	0
4	0

6.2 (5 points). Tired of always doing Lagrange interpolation by hand, you stumble across someone's (poorly documented) code to do it for you. However, to your frustration, you find that it has an off-by-one error! For example, if you give it the points $(1, 3)$, $(2, 2)$, and $(3, 4)$, it would instead interpolate the points $(1, 4)$, $(2, 3)$, and $(3, 5)$ (formally, when asked to interpolate a polynomial over $GF(p)$ through some point (x, y), it will instead interpolate a polynomial through $(x, y + 1)$).
But being the clever hacker that you are, you realize that, without touching any of the code itself, for any $g(x)$ that the program returns, you can find a new polynomial $p(x)$ that is the one you originally wanted.

Describe how to construct $p(x)$ using $g(x)$. Justify your answer.

(Specifically, we are asking: given a polynomial $g(x)$ of degree $n-1$ over $GF(p)$ through points $(x_1, y_1 + 1), (x_2, y_2 + 1), \ldots, (x_n, y_n + 1)$, find a polynomial $p(x)$ of degree $n - 1$ over $GF(p)$ through points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$).

Answer:

$$p(x) \equiv g(x) - 1 \pmod{p}$$

For ever point x_i, we know $g(x_i) \equiv y_i + 1 \pmod{p}$. By subtracting 1 from both sides we can see that for every x_i, $g(x_i) - 1 \equiv y_i \pmod{p}$. So if we take the polynomial $g(x)$ and subtract 1 from it we are guaranteed that it will pass through every (x_i, y_i).

6.3 (10 points). After fiddling with the program in part 6.2 you end up with the following quadratic (degree 2) polynomials over $GF(p)$:

$w_1(x)$ which passes through the points $(1,3),(2,1),$ and $(3,1)$

$w_2(x)$ which passes through the points $(1,1),(2,2),$ and $(3,1)$

$w_3(x)$ which passes through the points $(1,1),(2,1),$ and $(3,4)$

Describe how to use these polynomials to find the quadratic polynomial $p(x)$ over $GF(p)$ that passes through the points $(1,3)$, $(2,2)$, and $(3,4)$. Express your answer in terms of $w_1(x)$, $w_2(x)$, and $w_3(x)$.

Answer: For each polynomial $w_1(x)$, $w_2(x)$, and $w_3(x)$ you can subtract 1 from each so now 2 points are the roots but it goes through the wrong y value for the x that is not a root. So you have to rescale each polynomial ie $w_1(x)$ goes through (1,3)(2,1)(3,1)

$w_1(x) - 1$ goes through (1,2)(2,0)(3,0)
$2^{-1}[w_1(x) - 1]$ goes through (1,1)(2,0)(3,0)
$3 * 2^{-1}[w_1(x) - 1]$ goes through (1,3)(2,0)(3,0)

$w_2(x) - 1$ goes through (1,0)(2,1)(3,0)
$2[w_2(x) - 1]$ goes through (1,0)(2,2)(3,0)

$w_3(x) - 1$ goes through (1,0)(2,0)(3,3)
$3^{-1}[w_3(x) - 1]$ goes through (1,0)(2,0)(3,1)
$4 * 3^{-1}[w_3(x) - 1]$ goes through (1,0)(2,0)(3,4)

Summing these functions we get:

$$p(x) = 3 * 2^{-1}[w_1(x) - 1] + 2[w_2(x) - 1] + 4 * 3^{-1}[w_3(x) - 1]$$