CS280, Spring 2008: Prelim Solutions

The test is out of 50; the points for each question are marked. Don't forget to put your name and student number on each blue book that you use. You can answer the questions in any order, but mark your work clearly. Don't forget to show all your work. Give us a chance to give you partial credit! You have 90 minutes. Good luck!

1. [4 points] Suppose $f : A \to A$ is defined by $f(x) = 3x - 5$. Is f a 1-1 and onto (i.e., injective and surjective) if $A = Z$? What about if $A = Q$ (the rational numbers)? (If you think f is injective or surjective, explain why. If you don't, give a counterexample.)

   Solution: f is injective in both cases: if $x \neq y$, then we must have $3x - 5 \neq 3y - 5$. f is surjective if $A = Q$: given a rational number y, then take $x = (y + 5)/3$. It is easy to see that x is rational, and $f(x) = y$. However, f is not surjective if the domain is Z. There is no integer x such that $3x - 5 = 2$.

2. [5 points]

   (a) [1 point] What is an *equivalence relation* on S. (It's enough to just list the three properties of an equivalence relation; you don't have to explain what they are.)

   (b) [4 points] Recall that the *reverse* of a relation R is $\{(b, a) : (a, b) \in R\}$. Let R^r be the reverse of R. Prove that, for any relation R, $R \cup R^r$ is symmetric.

   Solution: (a) An equivalence relation is a relation the is reflexive, symmetric, and transitive.

   (b) Suppose that $(x, y) \in R \cup R^r$. Then $(x, y) \in R$ or $(x, y) \in R^r$. If $(x, y) \in R$, then $(y, x) \in R^r$, so $(y, x) \in R \cup R^r$. If $(x, y) \in R^r$, then $(y, x) \in R$, so $(y, x) \in R \cup R^r$. Either way, $(y, x) \in (R \cup R^r)$, so $R \cup R^r$ is symmetrc. To get full credit, you had to explicitly show that if $(x, y) \in R \cup R^r$, then so is (y, x). It wasn't enough to just say that because $(x, y) \in R$, we must have $(y, x) \in R^r$. (Indeed, this is not a complete argument.)

3. [5 points] Consider the following algorithm:

   Input m $\quad [m \in N]$
   $k \leftarrow m$
   $n \leftarrow 0$
   while $k \neq 0$ **do**
   $\quad n \leftarrow n + m$
   $\quad k \leftarrow k - 1$
   end
   return n

(a) What does this algorithm compute on input m?

(b) Prove your claim in (a) by using an appropriate loop invariant and induction. (Note that you also have to prove that the algorithm terminates.)

Solution: This algorithm computes m^2 on input m. To prove this, we need an appropriate invariant. Let $P(N)$ be the statement "if $N \leq m+1$, then at the beginning of the Nth iteration of the loop with input m, we have $n = (N-1)m$ and $k = m-(N-1)$. We prove $P(N)$ by induction for $N \geq 1$. For the base case, $l = 1$. At the beginning of the first iteration, we have $k = m$ and $n = m$, as desired. Suppose that $P(N)$ holds; we prove $P(N+1)$. If $N+1 > k+1$, there is nothing to prove. If $N+1 \leq m+1$, by the induction hypothesis, at the beginning of the Nth iteration, $k = m-(N-1)$ and $n = (N-1)m$. During the next loop, k is decremented by 1 and m is added to n, so at the beginning of the $(N+1)$st iteration, we have $k = m-N = m-(N+1-1)$ and $n = mN = m(N+1-1)$, as desired. It follows from the induction that at the beginning of the $(m+1)$st iteration of the loop, we have $k = 0$ and $n = m^2$. It follows that, at this point, the program terminates and outputs m^2. Thus, we have proved both that the program terminates and that it computes m^2. Some common mistakes included people trying $P(m) = m^2$ rather than inducting on number of iterations with m fixed Don't forget that when you're looking for a loop invariant to try to prove a program correct, the m in $P(m)$ is almost always the number of times through the loop.

4. [3 points] Explain carefully what the bug is in the following argument:

> We prove by strong induction that all orders of fish for at least 10 pounds of fish can be filled using only 5-pound fish. Let $P(n)$ be the statement that an order of fish for n pounds of fish can be filled using only 5-pound fish. We prove $P(n)$ for $n \geq 10$. Clearly $P(10)$ is true: an order for 10 pounds of fish can be proved using two 5-pound fish. Suppose that $P(10), \ldots, P(n)$ are all true. We prove $P(n+1)$. We want to show that we can fill an order for $n+1$ pounds of fish using 5-pound fish, if $n \geq 10$. By the induction hypothesis, we can fill an order for $n-4$ pounds of fish using 5-pound fish. Add one more 5-pound fish, and we've filled the order for $n+1$ pounds. This completes the induction argument.

Solution: If $n-4 < 10$, then we can't apply the induction hypothesis, because $n-4$ is not between 10 and n. In particular, the argument fails if $n = 11$, because $n-4 = 7$. Note that it wasn't enough to say that the argument doesn't work if $n = 11$. You had to explain *why* it didn't work.

5. Suppose the sets $P_0, P_1, P_2, \ldots$ of bit strings (that is, strings of 0s and 1s) are defined inductively by taking $P_0 = \{\lambda\}$ (where λ denotes the empty string, which is taken to have length 0) and $P_{n+1} = P_n \cup \{x00, x01, x10, x11 : x \in P_n\}$. Let $P = \cup_{k=0}^{\infty} P_k$.

Let Q be the smallest set such that

- $\lambda \in Q$;
- if $x \in Q$, then $x00, x01, x10, x11 \in Q$.

(a) [5 points] Prove that P_n consists of all strings of even length at most $2n$.

(b) [6 points] Prove that $P = Q$.

Solution: For part (a), let $Q(n)$ be the statement "P_n consists of all strings of length $2k$ for $k \leq n$". We prove $Q(n)$ for $n \geq 0$ by induction. $Q(0)$ says that P_0 consists of all strings of length 0. which is clearly true, since λ is the only string of length 0, and $P_0 = \{\lambda\}$. Suppose that $Q(n)$ is true. To prove $Q(n+1)$, note that $P_{n+1} = P_n \cup \{x11, x01, x10, x11 : x \in P_n\}$. If $x \in P_n$, then either $|x| < 2n$, in which case it follows from the induction assumption that $\{x00, x01, x10, x11\} \subseteq P_n$ (since P_n consists of all strings of length $2k$ for $k \leq n$) or $|x| = 2n$, in which case $\{x00, x01, x10, x11\} \cap P_n = \emptyset$ (since each of $x00$, $x01$, $x10$, and $x11$ have length $2n+2$. Since P_n includes all strings of length $2n$, it follows that $P_{n+1} - P_n = \{x00, x01, x10, x11 : |x| = 2n\}$. Thus, $P_{n+1} - P_n$ consists of all strings of length $2(n+2)$ (since every string of length $2n+2$ has the form $x00$, $x01$, $x10$, or $x11$ for some string x with $|x| = 2n$. It follows that P_{n+1} consists of all strings of length $2k$ for $k \leq n+1$. This completes the induction. Note that many people proved that all the strings in P_{n+1} were of even length, and that they all had length at most $2n+2$, but forgot to prove that P_{n+1} consisted of *all* strings of length at most $2n+2$. You lost one point if you didn't show this. (That's what the "ALL" on some homeworks means.) A few people were confused about the difference between $|P_n|$ (which is the number of elements in P_n, which can be shown to be $(4^{n+1} - 1)/3$, and the lengths of the strings in P_n.

For part (b), we show that $P \subseteq Q$ and $Q \subseteq P$. To show that $Q \subseteq P$, it suffices to show that P satisfies the properties that characterize Q. Clearly $\lambda \in P_0 \subseteq P$. Moreover, if $x \in P$, then $x \in P_n$ for some n, so $\{x00, x01, x10, x11\} \subseteq P$. Since Q is the smallest set that satisfies these two properties, we must have $Q \subseteq P$. (If you just said $Q \subseteq P$ without saying why—namely, that Q is the *smallest* with these properties—you lost a point.)

To show that $P \subseteq Q$, it suffices to show that $P_k \subseteq Q$ for all k, since $P = \cup_{k=0} \infty P_k$. We do this by induction. Let $R(n)$ be the statement "$P_n \subseteq Q$". Clearly $P_0 \subseteq Q$, since, by assumption $\lambda \in Q$. Suppose that $P_n \subseteq Q$. To see that $P_{n+1} \subseteq Q$, suppose that $x \in P_{n+1}$. Either $x \in P_n$, in which case $x \in Q$ by the induction assumption, or there exists $x' \in P_n$ such that either $x = x'00$, $x = x'01$, $x = x'10$, or $x = x'00$. By the induction hypothesis, $x' \in Q$. By the second clause in the characterization of Q, $x \in Q$. Thus, $P_{n+1} \subseteq Q$.

6. [4 points] Bob and Alice want to choose a secret key that they can use for cryptography, but all that they have is a bugged phone line, so an eavesdropper can listen to all their messages. Bob proposes that they each choose a secret number. Call Alice's number a and Bob's number b. They also choose, over the telephone number, a prime p and another number q. (Don't worry about exactly how they choose a, b, p, and q. All that matters is that they discuss p and q over the telephone line, so that someone bugging the line will hear what they are.) Bob will then send Alice $bq \bmod p$, and Alice will send Bob $aq \bmod p$. Their key (which they will keep secret) is then $abq \bmod p$. Note that they can both compute $abq \bmod p$. Alice hears $bq \bmod p$ from Bob, and knows a (that's her secret) so she can compute $abq \bmod p$. Similarly, Bob knows b and $aq \bmod p$, so can compute $abq \bmod p$. As Bob explains, their wiretapper will know p, q, $aq \bmod p$, and $bq \bmod p$, but will not know a or b, so their key will be safe (that is, despite hearing all their telephone discussions, an eavesdropper will not be able to figure out $abq \bmod p$.

   Is Bob right? Explain why or why not.

   Solution: Bob is wrong. Suppose that $aq \bmod p = k_1$ and $bq \bmod p = k_2$. Knowing k_1,, k_2, q, and p, an eavesdropper can "solve" $aq \equiv k_1 \pmod{p}$ and $bq \equiv k_2 \pmod{p}$. That is, although the eavesdropper cannot compute a and b exactly, as we showed in class, she can compute $a \bmod p$ and $b \bmod p$. That enough for her to compute $abq \bmod p$.

   This argument is an indication of how subtle it can be to prove cryptographic protocols correct. Here's a variant of Bob's suggestion that is believed to be safe: Alice sends Bob $q^a \bmod p$ and Bob sends Alice $q^b \bmod p$. They can both then compute $q^{ab} \bmod p$ and use that for their secret key. The effectiveness of this method depends on the assumption (believed to be true) that computing $a \bmod p$ is hard, given p, q and $q^a \bmod p$.

7. [5 points] Use the Extended Euclid's algorithm to find s and t such that $42s+47t = 1$. (You *must* show the steps of the algorithm. It is not enough to just write down s and t.)

   Solution: We first do the gcd computation, then work backwards to find s and t. $\gcd(42, 47) = \gcd(42, 5) = \gcd(5, 2) = \gcd(2, 1) = 1$. Now working backwards,

$$\begin{aligned} 1 &= 2 - 1 \\ &= 2 - (5 - 2 \times 2) \\ &= 3 \times 2 - 5 \\ &= 3 \times (42 - 8 \times 5) - 5 \\ &= 3 \times 42 - 25 \times 5 \\ &= 3 \times 42 - 25 \times (47 - 42) \\ &= 28 \times 42 - 25 \times 47 \end{aligned}$$

8. [4 points] Show that if n is a composite and $\gcd(n, b) > 1$ then $b^k \not\equiv 1 \pmod{n}$ if $k \geq 1$. Hint: what can you say if $b^k \equiv 1 \pmod{n}$?

   Solution: Suppose that $\gcd(n, b) = q > 1$. If $b^k \equiv 1 \pmod{n}$, then $n \mid (b^k - 1)$. Since $q \mid n$, by the transitivity of divisibility, $q \mid (b^k - 1)$. Since $q \mid b$, we must have $q \mid b^k$. It follows that $q \mid 1$, contradicting the assumption that $q \neq 1$. A few comments on solutions: Some people used the "rule" that if $a \mid (b - c)$, then $a \mid b$ and $a \mid c$. This is false (for example, $3 \mid (4 - 4)$, but it is not the case that $3 \mid 4$). Others used the rule that if $b^k \equiv 1 \pmod{n}$, then b^k must be relatively prime to n. This is true, but for full credit, you needed to prove it. Some people said that if $b^k \equiv 1 \pmod{n}$, then $b \equiv 1 \pmod{n}$. This is false ($3^2 \equiv 1 \pmod{8}$, but $3 \not\equiv 1 \pmod{8}$). Finally, some people tried to apply Fermat's Little Theorem. This can't be applied here, since n is not necessarily prime.

9. [4 points] Recall that the ISBN number for a book is given by 10 numbers $a_1, \ldots, a_{10}$ where $a_1, \ldots, a_9$ are in the range 0–9 (inclusive) and a_{10} is in the range 0-10 (in practice, "10" is represented by an x) such that

$$(1 \times a_1) + (2 \times a_2) + \cdots (9 \times a_9) + (10 \times a_{10}) \equiv 0 \pmod{11}. \tag{1}$$

   Show that ISBN numbers can detect errors in single digits. More precisely, suppose that the ISBN number of a book is supposed to be $a_1, \ldots, a_{10}$ (so that these numbers satisfy Equation (**??**)), and a_3 is mistyped as a_3'. Show that a_1, a_2, a_3', a_4, $\ldots$, a_{10} can't be a valid ISBN number. That is, show that it cannot satisfy Equation (**??**). (The same argument works if any other digit is mistyped, of course.)

   Solution: If a_1, a_2, a_3', a_4, $\ldots$, a_{10} is a valid ISBN number, then

$$1 \times a_1 + 2 \times a_2 + 3 \times a_3' + \cdots + 9 \times a_0 + 10 \times a_{10} \equiv 0 \pmod{11}.$$

   Since $a_1, a_2, \ldots, a_{10}$ is also valid, we must have

$$1 \times a_1 + 2 \times a_2 + 3 \times a_3 + \cdots + 9 \times a_0 + 10 \times a_{10} \equiv 0 \pmod{11}.$$

   Thus,

$$1{\times}a_1{+}2{\times}a_2{+}3{\times}a_3'{+}\cdots{+}9{\times}a_0{+}10{\times}a_{10} \equiv 1{\times}a_1{+}2{\times}a_2{+}3{\times}a_3{+}\cdots{+}9{\times}a_0{+}10{\times}a_{10} \pmod{11}.$$

   It follows that $3(a_3' - a_3) \equiv 0 \pmod{11}$. Thus, $11 \mid 3(a_3' - a_3)$. Since 3 and 11 are relatively prime, this means that $11 \mid (a_3' - a_3)$. But since a_3' and a_3 are both between 0 and 9, $a_3' - a_3$ is between -9 and 9. The only number in that range divisible by 11 is 0. Thus, $a_3' = a_3$, contradicting the assumption that a_3 was mistyped. Some people tried to make essentially this argument without using formulas, and often lost a point or so for being a bit too fuzzy.

10. [5 points] Use the Chinese Remainder Theorem to find *all* integers x that satisfy the following system of congruences:

$$x \equiv 2 \pmod 3$$
$$x \equiv 1 \pmod 4$$
$$x \equiv 2 \pmod 7$$

Solution: First find y_1 such that $y_1 \equiv 2 \pmod 3$, $y_1 \equiv 0 \pmod 4$, and $y_1 \equiv= 0 \pmod 7$. As shown in class, y_1 must have the form $28y_1'$. Since $28 \equiv 1 \pmod 3$, it's easy to see that we can take $y_1' = 2$, so $y_1 = 56$.

Next find y_2 such that $y_2 \equiv 0 \pmod 3$, $y_2 \equiv 1 \pmod 4$, and $y_2 \equiv= 0 \pmod 7$. Thus, y_2 must have the form $21y_2'$. Since $21 \equiv 1 \pmod 4$, we can take $y_2' = 1$ and $y_2 = 21$.

Finally, we need to find y_3 such that $y_3 \equiv 0 \pmod 3$, $y_3 \equiv 0 \pmod 4$, and $y_3 \equiv= 2 \pmod 7$. y_3 must have the form $12y_3'$. Since $12 \equiv 5 \pmod 7$, and $6 \times 5 = 30 \equiv 2 \pmod 7$, we can take $y_3' = 6$ and $y_3 = 72$.

Finally, we can take $x = y_1 + y_2 + y_3 = 56 + 21 + 72 = 149$. This is one solution, but the problem asks you to give them all. If x is a solution, then all solutions have the form $x + kM$, where M in this case is $3 \times 4 \times 7 = 84$. Thus, the set of all solutions is $\{149 + 84k : k \in Z\}$.

[Grading: you got 2 points if you guessed the answer right but the work didn't reflect a knowledge of the Chinese Remainder Theorem. If you found a solution but did not give all the solutions, you got 5.]