CPSC 121 Midterm 2
Monday, November 7th, 2011

SAMPLE SOLUTIONS

[6] 1. Prove or disprove each of the following statements. (Your proof/disproof should be *very* brief!)

[3] a. $\forall x \in \mathbf{Z}, (x > 18 \wedge x < 22 \wedge \mathrm{Odd}(x)) \rightarrow \mathrm{Prime}(x)$.

Solution : The statement is FALSE. Consider $x = 21$. It's between 18 and 22 and odd, but it's not prime (divisible by 7 and 3).

[3] b. $\forall x \in \mathbf{Z}, (x > 2 \wedge x < 6 \wedge \mathrm{Odd}(x)) \rightarrow \mathrm{Prime}(x)$.

Solution : The statement is TRUE. 3 and 5 are the only odd numbers between 2 and 6, and both are primes.

[7] 2. Consider the following theorem: Any real number can be added to some real number to equal 121.

[2] a. Translate this theorem into predicate logic.

Solution : $\forall x \in \mathbf{R}, \exists y \in \mathbf{R}, x + y = 121$.

[5] b. Prove the theorem. (Your proof should be in the style we have learned in CPSC 121, though it will likely be brief.)

Solution : WLOG, let x be a real number. Let $y = 121 - x$. Then, $x + y = x + (121 - x) = 121$. QED
(Note: you needn't have stated this, but we know y is real since the real numbers are closed under addition and multiplication, and 121, x, and -1 are all real.)

[3] 3. Move the negation on the following statement "inward" as much as possible. When you're done, negation(s) should appear *only* on predicates—e.g., $\sim M(a, b)$ or $b \neq c$—and *not* on quantifiers or parenthesized expressions.
$\sim \exists a \in \mathbf{Z}, \mathrm{Foo}(a) \wedge (\forall b \in \mathbf{Z}^+, ab < a + b)$.

Solution : $\forall a \in \mathbf{Z}, \sim \mathrm{Foo}(a) \vee (\exists b \in \mathbf{Z}^+, ab \geq a + b)$.

[3] 4. Apply the contrapositive equivalence rule to the following statement. Again, in your result, move all negations inward as much as possible.
$\forall x \in D, (\exists y \in E, P(x, y) \wedge Q(y)) \rightarrow R(x)$.

Solution : $\forall x \in D, \sim R(x) \rightarrow (\forall y \in E, \sim P(x, y) \vee \sim Q(y))$.

[8] 5. Let V be the set of voters. Let C be the set of candidates. Let $\mathrm{Prefers}(v, c_1, c_2)$ mean voter v prefers candidate c_1 to candidate c_2. Let $\mathrm{Beats}(c_1, c_2)$ mean candidate c_1 beats candidate c_2 in the election. (Do not assume $\mathrm{Prefers}(v, c, c)$ or $\mathrm{Beats}(c, c)$ is always false for any voter v and candidate c.)

[4] a. Translate this statement to predicate logic: If every voter prefers one candidate to a second *different* candidate, then the second one cannot beat the first in the election.

Solution :

$$\forall c_1 \in C, \forall c_2 \in C, (c_1 \neq c_2 \wedge \forall v \in V, \mathrm{Prefers}(v, c_1, c_2)) \rightarrow \sim \mathrm{Beats}(c_2, c_1).$$

(Note: The order of the neighbouring $\forall$ quantifiers doesn't matter. The $\forall$ on v must be inside the antecedent of the conditional—although it can move outside $c_1 \neq c_2$—*or* must be changed to an existential. Also, moving the quantifier on v outside of the quantifiers on c_1 or c_2 makes for a poorer translation.)

[4] b. Define a predicate $\mathrm{Mid}(v, c)$ meaning voter v prefers some *other* candidate to c but also prefers c to some *third* candidate.

Solution :

$$\mathrm{Mid}(v, c) \equiv \exists c_1 \in C, \exists c_2 \in C, c_1 \neq c_2 \wedge c_2 \neq c \wedge c_1 \neq c \wedge \mathrm{Prefers}(v, c_1, c) \wedge \mathrm{Prefers}(v, c, c_2).$$

(Note: The order of the neighbouring $\exists$ quantifiers doesn't matter. We do need all three $\neq$ expressions to make these three separate candidates.)

[2] 6. In proving the following theorem with direct proof techniques, you would choose values for y and z. Which of a, b, and c can y's and z's values depend on?

Theorem: $\forall a \in A, \forall b \in B, \exists y \in Y, P(a, b, y) \wedge \forall c \in C, (Q(y, c) \rightarrow \exists z \in Z, R(b, c, z, y))$.

Solution : y's value can depend on only a and b. (Not c because the quantifier on c is inside the quantifier on y.)

Solution : z's value can depend on all three of a, b, and c.

[5] 7. For each of the following theorems, write the letter of **EVERY** method from the following list that could be a promising **FIRST** step in proving the theorem.

List of methods:

(A) Witness proof (constructive/non-constructive proof of existence)

(B) Exhaustive proof.

(C) "Without loss of generality" proof (generalizing from the generic particular)

(D) Antecedent assumption.

(E) Proof by contradiction.

_____ $(\forall x \in D, P(x)) \rightarrow q$

Solution : Promising approaches: D and E. (Contradiction is a reasonable first step for **all** of the theorems.)

_____ $\forall z \in \mathbf{R}^+, \exists y \in \mathbf{R}, y < z \wedge M(y, z)$.

Solution : Promising approaches: C and E. Exhaustive proof is *not* a promising strategy for the infinite domain of positive real numbers.

_____ $\exists q \in S, K(q) \vee \sim R(q)$.

Solution : Promising approaches: A and E.

_____ $\forall x \in C, \forall y \in D, A(x, y)$, where C is the set of students in CPSC 121 this term.

Solution : Promising approaches: B, C, and E. Exhaustive proof *is* promising since there is a finite (and not *that* large) number of 121 students this term.

[6] 8. For each of the following theorems, indicate the most complete "direct" proof approach—no proof by contradiction, no use of logical equivalences—that you can by writing the letters of the techniques from the previous problem in the order you would use them. Ignore any blanks you don't need.

So, for a proof that starts as a witness proof, then uses "without loss of generality" twice, and then antecedent assumption: write A in blank #1, C in blanks #2 and #3, D in blank #4, and nothing in blank #5.

Theorem: $\forall z \in \mathbf{Z}^+, \exists p \in S, Q(z, p)$.

Solution : #1 C, #2 A

Theorem: $\forall x \in \mathbf{Z}, (\exists y \in \mathbf{Z}, \text{Froogly}(x, y)) \rightarrow (\exists z \in \mathbf{Z}^+, \text{Froogly}(x, z) \rightarrow x > z)$.

Solution : #1 C, #2 D, #3 A, #4 D

Note, we assume $\exists y \in \mathbf{Z}, \text{Froogly}(x, y)$. We do not prove it. So, we certainly wouldn't use a proof technique on its quantifier.

Theorem: If the product of two integers has the "*silly nonsense words*" property, then the two integers are *flibbergybooooodifilcullayploo*.

Solution : #1 C, #2 C, #3 D

[4] 9. After a traumatic head injury—received while working as an international logic spy—you discover the following partial proof on the back of your hand. Write out the theorem the proof addresses **in as much detail as possible**.

> Without loss of generality, let x be a positive integer. We now consider the contrapositive of the remaining theorem. Assume $R(x)$ holds. Let $y = 5x$. We now show that $S(x, y)$ holds but $S(y, x)$ does not. We proceed by... [*further notes obscured by either blood or ketchup*]

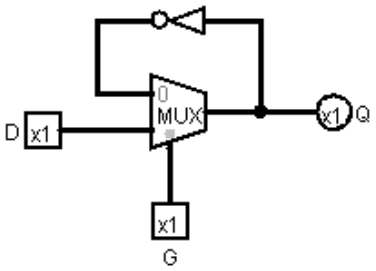
Solution :

$\forall x \in \mathbf{Z}^+, (\sim \exists y \in \mathbf{Z}^+, S(x, y) \wedge \sim S(y, x)) \rightarrow \sim R(x)$

Note: We don't really know the domain of y (except that it must include at least the positive integers that are multiples of 5); so, just $??$ would be fine there. The parentheses around the quantifier on the left-hand side are necessary, although they could be inside the negation. Otherwise, it would appear that the negation and quantifier apply to the entire conditional, not just its antecedent. It would be fine to move the negation on the quantifier inward (though the quantifier must still be in parentheses!) as long as it was done correctly.

[4] 10. We used a multiplexer to build a latch, a circuit capable of loading and storing a value.

Consider the following proposed design for a different latch. The latch's intended semantics are: (1) When G is 1, the latch loads a new value from D (outputting that value). (2) When G is 0 the latch stores the *negation* of the value it loaded from D.

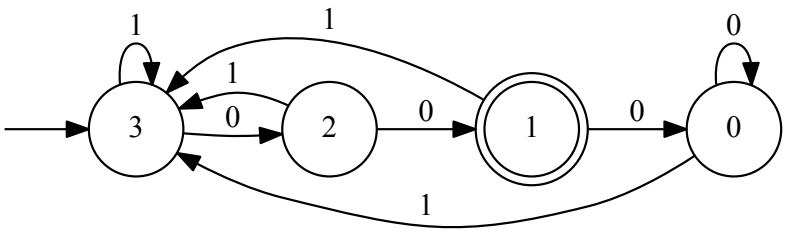


Does this circuit work correctly? Briefly justify your answer.

Solution : No, this circuit does not work correctly. When G is 1, the circuit works fine. When G is 0, the circuit is unstable, oscillating between 0 and 1, *not* just storing the negation of the data it loaded.

(It would certainly be possible to build this circuit with *two* multiplexers, both controlled by G. The first is just a D latch. The second outputs the D latch's value when G is 0 but the negation of the D latch's value when G is 1.)

[13] 11. Consider the following DFA:



[2] a. Circle **all** of the following inputs that will be accepted by this DFA:

the empty string 0100 1011000 00 1 000000 0000001

Solution : Only 0100 and 00 are accepted.

[3] b. Clearly and concisely describe the language accepted by this DFA.

Solution : Accepts strings that end with exactly two consecutive 0s (i.e., neither with three or more 0s in a row nor with fewer than two).

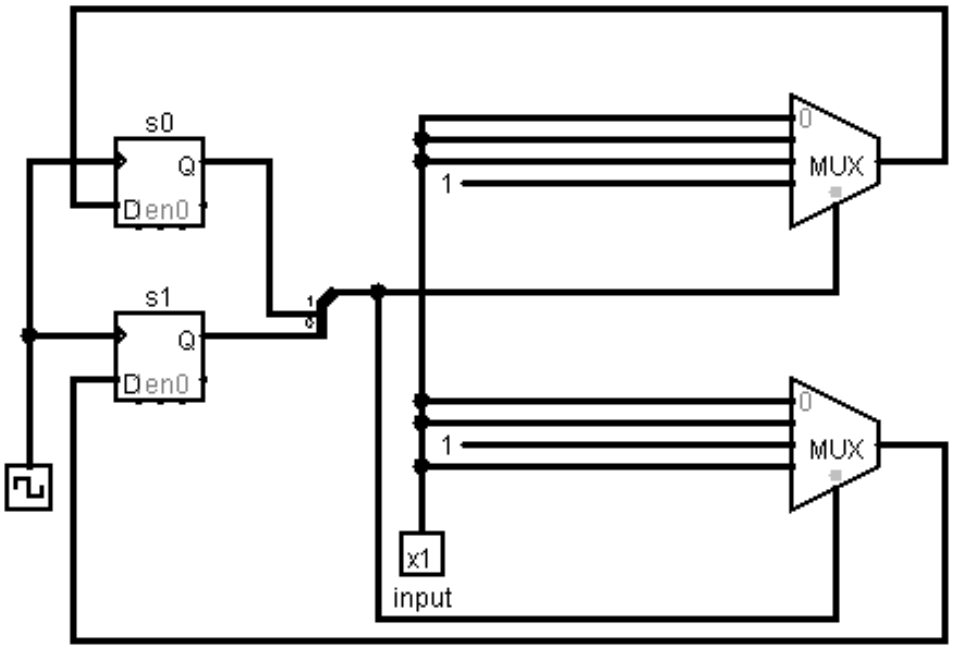
[2] c. Fill in the following truth table indicating what the next state of the DFA should be after receiving an input in state 2. **For this and subsequent parts**, s_0 is the first (left-most) bit of the state number. s_1 is the second (right-most) bit of the state number.

Solution :

input	s_0	s_1
1	1	1
0	0	1

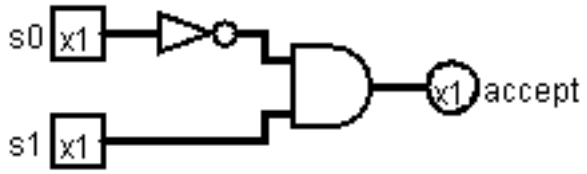
[4] d. Complete the following circuit implementing this DFA. Note that we have implemented circuits for states 0 and 1; you need only complete states 2 and 3.

Solution :

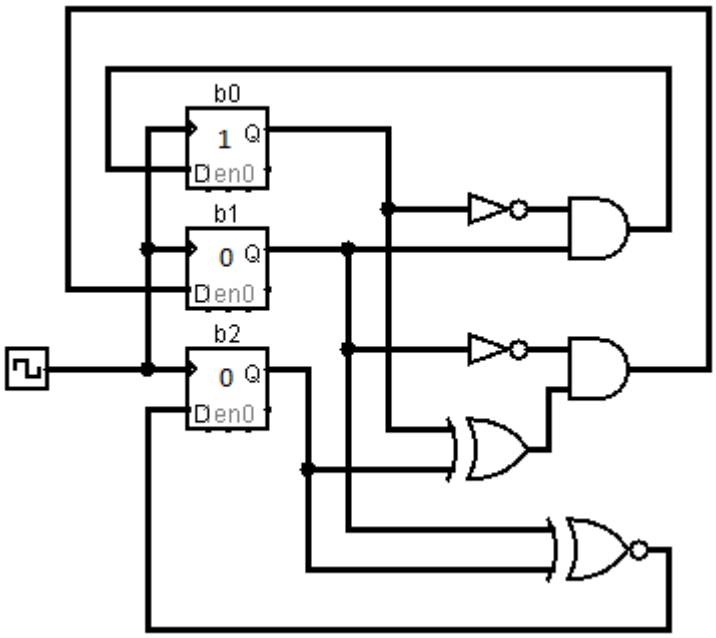


[2] e. Finally design a circuit that, given the state of the DFA as input, determines whether the DFA is in an accepting state.

Solution : This is a combinational circuit design problem.



[6] 12. Consider the following sequential circuit that stores an unsigned 3-bit value:



As shown, the circuit stores the number 4, with $b_0 = 1, b_1 = 0, b_2 = 0$. Complete the following table indicating what state the circuit will be in for the next three clock ticks. The table shows how the circuit reached the point where it stores 4, starting from storing 0.

Solution :

After 0 ticks:	0
After 1 tick:	1
After 2 ticks:	2
After 3 ticks:	4
After 4 ticks:	**3**
After 5 ticks:	**5**
After 6 ticks:	**0**

This happens to repeatedly generate a (somewhat arbitrary) permutation of the numbers 0–5. It came from a bug in a circuit meant to start at 1 and repeatedly multiply by 2 in (mod 5)—clock arithmetic with 5 ticks on the clock. Oops! :)

[13] 13. Prove the following theorem: If an integer greater than 1 divides two positive integers a and b, then a and b both divide some integer less than ab (the product of a and b).

Your proof should be in the style we have learned in CPSC 121.

Reminder: p divides q exactly when there is an integer k such that $pk = q$.

Solution : Short version. Assume an integer $d > 1$ divides both a and b. So, $d = ak_1$ and $d = bk_2$ for some pair of integers k_1 and k_2. Note that $dk_1k_2 = ak_2$ and $dk_1k_2 = bk_1$. So, a and b both divide dk_1k_2, and $dk_1k_2 < d^2k_1k_2 = ab$ (since $d > 1$). QED

Long version. Without loss of generality, let a and b be positive integers. Assume some integer $d > 1$ divides both a and b. So, we know there are integers k_1 and k_2 such that $dk_1 = a$ and $dk_2 = b$. We need to prove there's an integer $m < ab$ such that $ak_3 = m$ and $bk_4 = m$ for some integers k_3 and k_4.

Let $m = dk_1k_2$, which is less than $d * dk_1k_2$ since $d > 1$. But, $d * dk_1k_2 = dk_1 * dk_2 = ab$.

Let $k_3 = k_2$. Let $k_4 = k_1$.

Then, $m = dk_1k_2 = ak_2 = ak_3$. So, a divides m.

And, $m = dk_1k_2 = bk_1 = bk_4$. So, b divides m.

QED

Where did the insight to look at dk_1k_2 come from? Well, we know $a = dk_1$ and $b = dk_2$, and we know we want to think about ab; so, we just try multiplying them together: $dk_1 * dk_2$. Clearly both dk_1 (i.e., a) and dk_2 (i.e., b) divide that number, but they would even if there were only one d. So, we give that a shot.

It would also work to look at dk_1 and dk_2 and ask yourself: what's a number these both divide? The smallest one we can be sure of is dk_1k_2.