CPSC 121: Models of Computation
Midterm Exam, 2009 February 24

Solution : SAMPLE SOLUTION

Name: _____ Student ID: _____

Signature: _____

- You have **90 minutes** to write the 10 questions on this quiz.
- A total of **75 marks** are available. You may want to complete what you consider to be the easiest questions first!
- Ensure that you clearly indicate a single legible answer for each question.
- You are allowed any reasonable number of textbooks and a single binder or folder of notes as references. Otherwise, no notes, aides, or electronic equipment are allowed.
- Good luck!

UNIVERSITY REGULATIONS

1. Each candidate must be prepared to produce, upon request, a UBCcard for identification.

2. Candidates are not permitted to ask questions of the invigilators, except in cases of supposed errors or ambiguities in examination questions.

3. No candidate shall be permitted to enter the examination room after the expiration of one-half hour from the scheduled starting time, or to leave during the first half hour of the examination.

4. Candidates suspected of any of the following, or similar, dishonest practices shall be immediately dismissed from the examination and shall be liable to disciplinary action:

- having at the place of writing any books, papers or memoranda, calculators, computers, sound or image players/recorders/transmitters (including telephones), or other memory aid devices, other than those authorized by the examiners;
- speaking or communicating with other candidates; and
- purposely exposing written papers to the view of other candidates or imaging devices. The plea of accident or forgetfulness shall not be received.

5. Candidates must not destroy or mutilate any examination material; must hand in all examination papers; and must not take any examination material from the examination room without permission of the invigilator.

6. Candidates must follow any additional examination rules or directions communicated by the instructor or invigilator.

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total

This page intentionally left blank.
If you put solutions here or anywhere other than the blank provided for each solution, you must *clearly* indicate which problem the solution goes with and also indicate where the solution is at the designated area for that problem's solution.

1 Propositional Logic Tautologies, Contradictions, and Contingencies [4 marks]

For each propositional logic statement below, indicate (by circling the appropriate answer) whether it is a tautology, a contradiction, or a contingency (neither a tautology nor a contradiction).

These are worth **[1 mark]** each.

1. $p \rightarrow (q \rightarrow \sim p)$

Circle one: **tautology** **contradiction** **contingency**

Solution : contingency

2. $(\sim(p \oplus q)) \leftrightarrow (\sim q \oplus p)$

Circle one: **tautology** **contradiction** **contingency**

Solution : tautology

3. $((p \leftrightarrow q) \leftrightarrow r) \rightarrow (p \vee q \vee r)$

Circle one: **tautology** **contradiction** **contingency**

Solution : tautology

4. $(p \vee \sim q) \wedge (q \vee \sim r) \wedge (r \vee \sim s) \wedge \sim(p \vee \sim s)$

Circle one: **tautology** **contradiction** **contingency**

Solution : contradiction

2 Circuit Properties [4 marks]

For each of the following statements, label them as **definitely true**, **possibly true**, or **definitely false** for any combinational circuit that can be modeled by one or more propositional logic statements. (That is, circle one of those options for each statement.)

Note: combinational circuits are the circuits we worked with in the first part of the course in which no path through the circuit loops back on itself.

These are worth **[1 mark]** each.

1. Every input to the circuit is connected to the input of a gate.

Circle one: **definitely true** **possibly true** **definitely false**

Solution : possibly true

2. Some gate's output is connected to the output of another gate.

Circle one: **definitely true** **possibly true** **definitely false**

Solution : definitely false

3. No input to the circuit is connected to the input of more than one gate.

Circle one: **definitely true** **possibly true** **definitely false**

Solution : possibly true

4. The number of gates is less than or equal to the number of outputs of the circuit.

Circle one: **definitely true** **possibly true** **definitely false**

Solution : possibly true

3 Representing Relationships with Predicate Logic [5 marks]

Let S be the set of all students and C be the set of all courses. Consider the following predicates:

- $Required(s, c)$ means course c is required for student s to graduate
- $Enrolled(s, c)$ means student s is enrolled in course c
- $Full(c)$ means course c has no room for further enrollment

1. Translate the following into predicate logic. **[2 marks]**

Any course that's full has at least one student enrolled in it.

Solution : $\forall c \in C, Full(c) \rightarrow \exists s \in S, Enrolled(s, c).$

The existential can be just inside the universal and, of course, various other equivalent solutions exist.

2. Imagine you were using proof by contradiction to prove this theorem:

$\exists c \in C, \exists s \in S, Enrolled(s, c) \vee Full(c).$

(a) State the assumption that would begin your proof (in predicate logic). Hint: any correct statement is acceptable; you need not simplify in any way. **[1 mark]**

Solution : $\sim \exists c \in C, \exists s \in S, Enrolled(s, c) \vee Full(c).$

Various equivalent solutions exists, but this one's easy to figure out!

(b) Given your assumption, what, if anything, would you know about any student and course, regardless of which particular student or course they are? (Write your answer in English, not predicate logic.) **[2 marks]**

Solution : The course is not full, and the student is not enrolled in the course.

4 Proof Critique [5 marks]

Let $\mathbf{R}$ be the set of all real numbers, let $\mathbf{Z}$ be the set of all integers and let $\mathbf{Z}^+$ be the set of all positive integers. The set of rational numbers, $\mathbf{Q}$, can be defined as the set of numbers $\frac{a}{b}$ for some $a, b \in \mathbf{Z}$ $(b \neq 0)$. Without loss of generality, one can further stipulate that a and b have no factors in common. That is, there is no number $k \in \mathbf{Z}^+, k > 1$ that divides both a and b. A real number $x \in \mathbf{R}$ that is not rational is called irrational. (Note that any integer is a rational number since an integer $n = n/1$.)

1. Assuming its premises are true, can a proof that $\sqrt{4}$ is irrational be valid? Why or why not? **[2 marks]**

Solution : Note: "is rational" in the question text was changed to "is irrational" after the exam.

No. $\sqrt{4} = 2$, which is an integer and therefore rational. So, no valid proof can prove it's irrational.

2. Indicate the step or steps in the proof below that are invalid or state here that the proof is valid. **[3 marks]**

Note that the proof makes use of the following lemma, which is valid.

Lemma: $\forall n \in \mathbf{Z}^+$, if n^2 is even then n is even.

Proof (by contradiction): $\sqrt{4}$ is irrational.

Assume $\sqrt{4}$ is rational (i.e., assume the negation of theorem).

Then $\sqrt{4} = \frac{a}{b}$ for some $a, b \in \mathbf{Z}$ $(b \neq 0)$.

Further, assume that a and b share no common factors. Now,

$$\begin{aligned} (\sqrt{4})^2 &= \left(\frac{a}{b}\right)^2 \\ 4 &= \frac{a^2}{b^2} \\ a^2 &= 4b^2 \end{aligned}$$

Hence, a^2 is even. If a^2 is even then a is even (by given lemma) and $a = 2c$ for some $c \in \mathbf{Z}$. Substituting back into $a^2 = 4b^2$, we obtain

$$\begin{aligned} (2c)^2 &= 4b^2 \\ 4c^2 &= 4b^2 \\ b^2 &= c^2 \end{aligned}$$

Hence, b^2 also is even. Once again, if b^2 is even then b is even (by given lemma).

Thus, a and b share the common factor 2 which contradicts our assumption that they have no common factors.

Therefore, $\sqrt{4}$ is irrational QED

Solution : Invalid because $b^2 = c^2$ gives no evidence that b^2 is even.

5 Number Representation [9 marks]

Since you do not have a calculator, you may leave your answers as arithmetic expressions (e.g., 53 + 27 + 5), but the correct answers should be obtainable from your expressions simply by entering them into a basic calculator.

For parts 1–3 below, consider the pattern of digits 11010001.

1. Could these digits be a signed binary integer? If so, assume they are and give their value in base 10. If not, explain why not. **[2 marks]**

Circle one: **YES** **NO**

If **YES**, give the base 10 value.
If **NO**, explain why not.

Solution : Yes: $-47 = -(32 + 8 + 4 + 2 + 1)$

2. Could these digits be an unsigned binary integer? **[1 mark]**

Circle one: **YES** **NO**

Solution : Yes, since they're only 1s and 0s. (The initial 1 doesn't stop this from being interpreted as unsigned.)

3. Could these digits be an integer in base 10? **[1 mark]**

Circle one: **YES** **NO**

Solution : Yes, since they're only 1s and 0s. (A number like 100 can certainly be a decimal number; so, can any number that contains only the digits 0–9.)

4. When using 8 bits, can we represent the most distinct values by interpreting the bits as an unsigned binary integer, as a signed binary integer, or as an unsigned binary fraction (with a 1/2s place, 1/4s place, . . . , and 1/256ths place)? **[3 marks]**

Circle the representation that yields the most distinct values. If there is a tie, circle all tied representations:

Unsigned binary integer **Signed binary integer** **Unsigned binary fraction**

Solution : All are tied. Each represents $2^8 = 256$ distinct values. The unsigned integer can represent 0–255, signed integer -128–127, and the fraction $\frac{0–255}{256}$.

5. Briefly explain one compelling reason why representing binary numbers in base 16 would be more convenient for humans than base 256. **[2 marks]**

Solution : Remembering 16 different symbols and their ordering is not too taxing for humans (especially since we've mostly already memorized the first 10). However, remembering 256 different symbols and their ordering would be tricky.

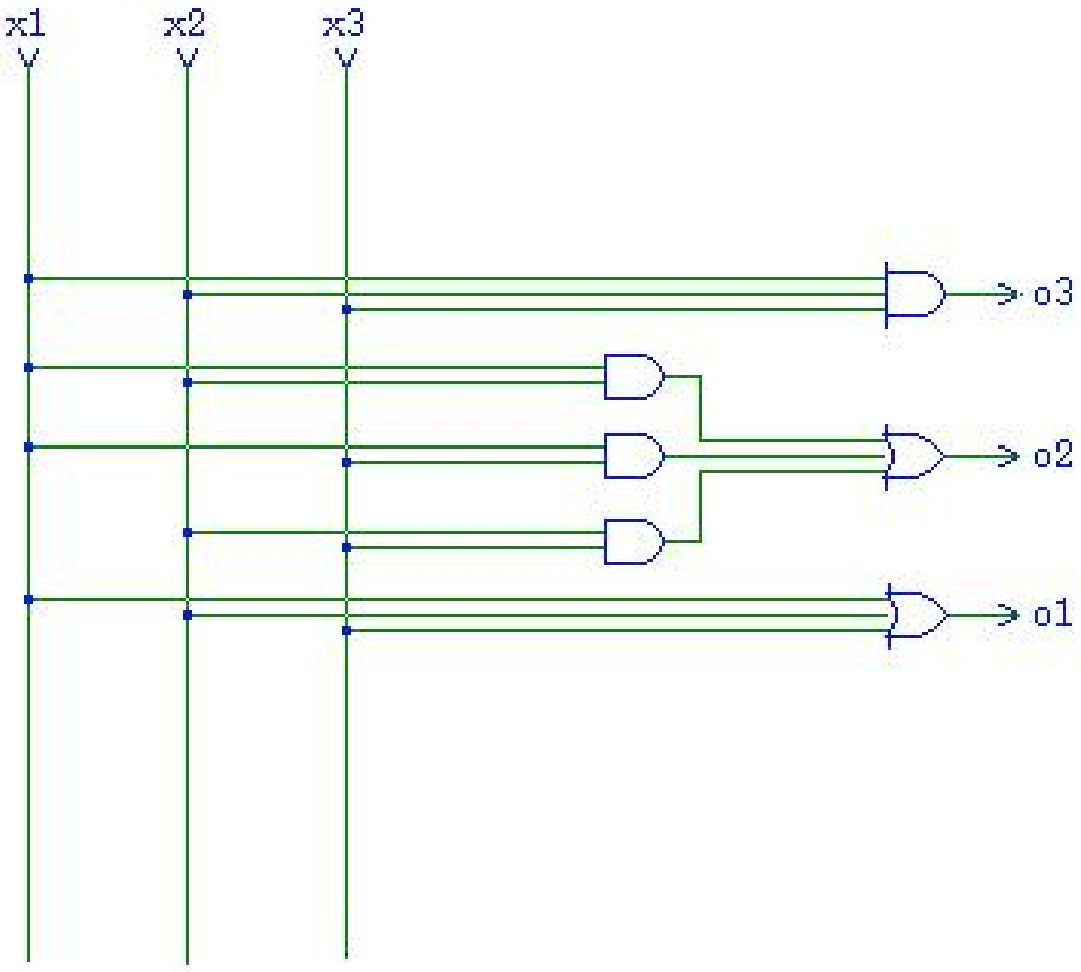
6 Circuit Design [17 marks]

Using the steps below, design a circuit that takes two 3-bit numbers as input and outputs the one with the most 1s in it. (That is, the three output bits are identical to and in the same order as the input with the most 1s. If both have the same number of 1s, the output may match either input.)

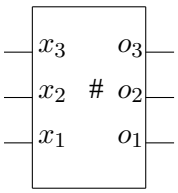
You may design your circuit using inverters, 2-input XOR and XNOR gates, 2- or 3-input AND, OR, NAND, and NOR gates, 2:1 multiplexers, and the modules described below.

1. Design a circuit that takes three inputs and produces three outputs, o_1, o_2, and o_3. o_1 is 1 if and only if one or more of the inputs are 1s. o_2 is 1 if and only if two or more of the inputs are 1s. o_3 is 1 if and only if all three of the inputs are 1s. **[5 marks]**

Solution : The insight is that o_1 is just the OR of the inputs, o_3 is the AND of the inputs, and o_2 is the OR of each of the three pairs of inputs AND'd together.



In the next part, use your circuit from part 1 as a module with the following symbol:



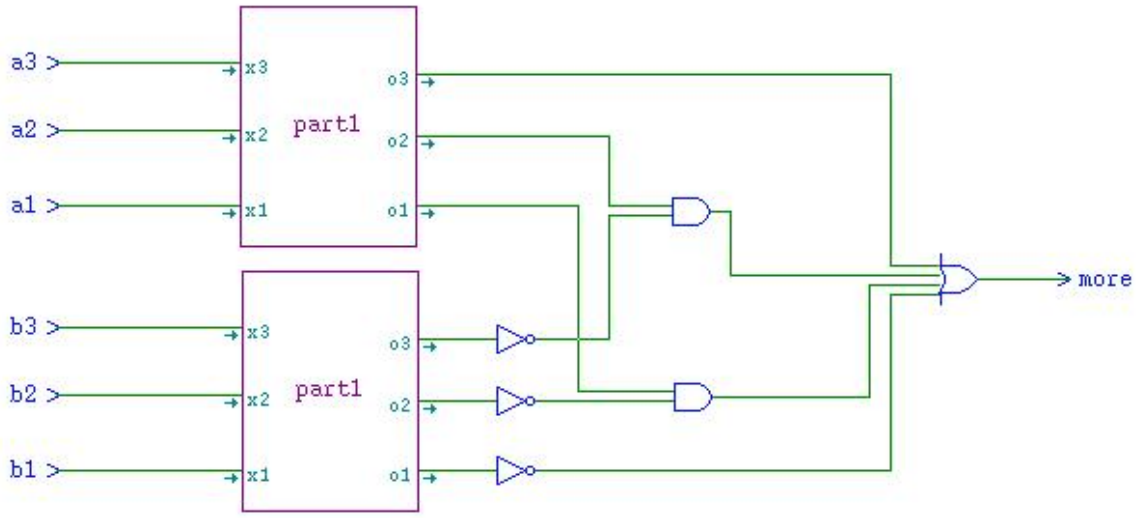
You should use this module in part 2, even if you do not complete part 1. Assume the module works as described in the problem statement for part 1.

2. Design a circuit that takes two 3-bit inputs and has a single output named $more$. The output is true if and only if the first input has at least as many 1s as the second. **[7 marks]**

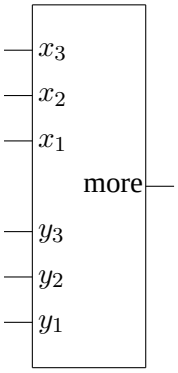
Solution : Note: the word "named" was added to the question text after the exam.

The insight is to run both the x and y inputs through the module from the previous part and then enumerate the four cases where x has more 1s than y: whenever x has three 1s, whenever x has two or more 1s but y doesn't have three, whenever x has one or more 1s but y doesn't have two, and whenever y doesn't have a 1. The first of these cases is an output of one of the modules; the last is the negation of a module output. The middle two are ANDs of an output from the x module and the negation of an output from the y module.

Our diagram varies slightly in naming conventions due to TkGate constraints.



In the next part, use your circuit from part 2 as a module with the following symbol:

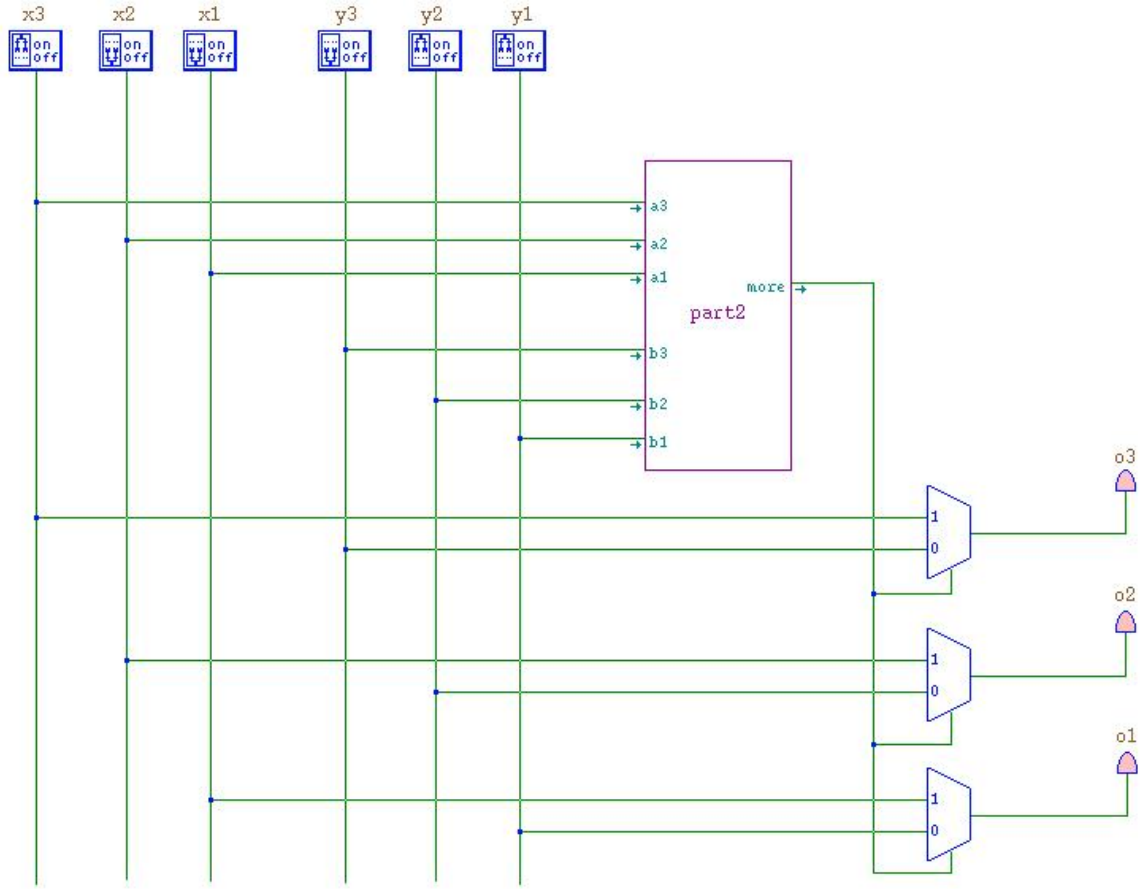


You should use this module in part 3, even if you do not complete part 2. Assume the module works as described in the problem statement for part 2. Hint: this is actually the easiest part, particularly if you use multiplexers; you may want to complete it first!

3. Design the full circuit that takes two 3-bit inputs and outputs the one with the most 1s. **[5 marks]**

Solution : The insight is to use the module from the previous part to determine which of x and y to pass through. Then, use three multiplexers (one for each matching bit of the x/y inputs) to perform the actual selection. The $more$ output of the module is the select input of the multiplexer.

Our diagram varies slightly in naming conventions due to TkGate constraints.



7 Proving Properties of Computing Systems [13 marks]

A decision algorithm is an algorithm that, when run on an input, answers either yes or no. Furthermore, there is a "correct" answer for each input, either yes or no, and the algorithm may or may not get the correct answer.

A decision algorithm is called sound exactly when: if the algorithm reports that the answer for an input is yes, the correct answer for that input is yes.

A decision algorithm is called complete exactly when: if the correct answer for an input is yes, then the algorithm reports that the answer for that input is yes.

(An algorithm that is not sound is "unsound". An algorithm that is not complete is "incomplete".)

Consider the following procedure to generate a new decision algorithm, which we call "zipping": the algorithm resulting from zipping decision algorithm a_1 and decision algorithm a_2 is called zip_{a_1,a_2} and answers yes exactly when one or both of a_1 and a_2 answer yes.

1. Prove *using a direct proof* that for any pair of algorithms a_1 and a_2, if a_1 and a_2 are both sound, zip_{a_1,a_2} is sound. **[4 marks]**

Solution : Assume a_1 and a_2 are both sound. Whenever zip_{a_1,a_2} answers yes, it does so because one (or both) of a_1 and a_2 answered yes. Whichever one answered yes, we know since it is sound that the correct answer to the problem was also yes. So, in every case that zip_{a_1,a_2} answers yes, the correct answer is yes, meaning zip_{a_1,a_2} is sound. QED.

2. Prove *using the contrapositive* that if zip_{a_1,a_2} is incomplete, then both a_1 and a_2 are incomplete. **[5 marks]**

Solution : We proceed by proving the contrapositive, that if either a_1 or a_2 is complete, then zip_{a_1,a_2} is complete.

Assume a_1 or a_2 is complete.

Consider the case when a_1 is complete. Anytime the correct answer is yes, a_1 answers yes. Anytime a_1 answers yes, zip_{a_1,a_2} also answers yes by construction.

The case when a_2 is complete behaves correspondingly.

Thus, anytime the correct answer is yes, zip_{a_1,a_2} answers yes, meaning zip_{a_1,a_2} is complete. QED.

3. Prove *using contradiction* that zipping an unsound algorithm with itself does not produce a sound algorithm. **[4 marks]**

Solution : We proceed by contradiction.

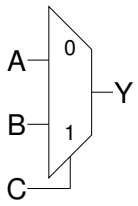
Assume for contradiction that zipping some unsound algorithm with itself does produce a sound algorithm. Let a be that unsound algorithm; so, $zip_{a,a}$ is the sound algorithm produced with zipping.

Since a is unsound, there's at least one input for which the correct answer is no, but to which a answers yes. By construction, $zip_{a,a}$ also answers yes to that input. However, since $zip_{a,a}$ is sound, it never answers yes to an input whose correct answer is no.

This is a contradiction; so, our assumption was false. QED.

8 Universality [10 marks]

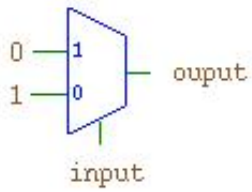
A 2 input, 1 select line MUX is referred to as a 2:1 MUX. The task is to prove that a 2:1 MUX is universal. For a 2:1 MUX, the output Y is equal to input A when C is 0 and equal to input B when C is 1. The gate symbol is:



Note: For each of the next three parts of this problem, you may supply any otherwise unused MUX input with constant 0 or constant 1 (i.e., wire an input to ground or power).

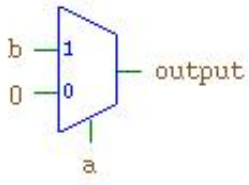
1. Show how a 2:1 MUX can be used to implement an inverter. **[2 marks]**

Solution : Use the inverter's input as the select input to the mux. Feed 0 to the "1" input of the mux and 1 to the "0" input of the mux (so that the output will always be opposite to the select line's value).



2. Show how a 2:1 MUX can be used to implement a 2-input AND gate. **[3 marks]**

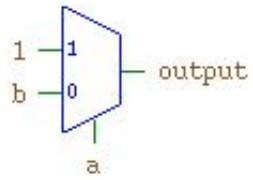
Solution : Use one of the AND's inputs as the select input to the mux. Feed the other input to the AND to the "1" input of the mux and 0 to the "0" input of the mux (so that the output will only ever be 1 if both inputs to the AND are 1).



3. Show how a 2:1 MUX can be used to implement a 2-input OR gate. **[3 marks]**

Solution : Use one of the OR's inputs as the select input to the mux. Feed the other input to the OR to the "0" input of the mux and 1 to the "1" input of the mux (so that the output will only ever be 0 if both inputs to the OR are 0).

Alternatively, since we've already established an implementation of inverters and AND in terms of multiplexers, just use the circuit for $\sim(\sim p \wedge \sim q) \equiv p \vee q$.



4. Does what we requested in steps 1–3—showing implementations of each gate in terms of a multiplexer—prove that the 2:1 MUX is universal? Why or why not? **[2 marks]**

Solution : No. We have shown the implementations but not proven their correctness. If we proved their correctness then, based on the premise that AND, OR, and NOT are universal, we have proven that the 2:1 MUX is as well.

9 A Bit More Number Representation [4 marks]

Consider the 8-bit unsigned binary number 00011011.

1. Translate it to decimal. **[2 marks]**

Solution : $16 + 8 + 2 + 1 = 27$

2. Translate it to hexadecimal. **[2 marks]**

Solution : $1B$

10 Abstract Proof [4 marks]

Describe in as much detail as possible what strategy you would use to prove the following theorem using a direct proof approach.

$\forall y \in E, \exists z \in F, P(y, z) \leftrightarrow R(y, z)$.

Solution : Without loss of generality, select an arbitrary y from the domain E. Pick a z, which may be based on y. Then, proceed with two proofs:

- Assume $P(y, z)$ and prove $R(y, z)$.
- Assume $R(y, z)$ and prove $P(y, z)$.

This page intentionally left blank.
If you put solutions here or anywhere other than the blank provided for each solution, you must *clearly* indicate which problem the solution goes with and also indicate where the solution is at the designated area for that problem's solution.

This page intentionally left blank.
If you put solutions here or anywhere other than the blank provided for each solution, you must *clearly* indicate which problem the solution goes with and also indicate where the solution is at the designated area for that problem's solution.

This page intentionally left blank.
If you put solutions here or anywhere other than the blank provided for each solution, you must *clearly* indicate which problem the solution goes with and also indicate where the solution is at the designated area for that problem's solution.