CSCI 1315
Discrete Mathematics for Computer Science
Assignment 2

Logic and Proof

Due: February 18th, 11:59PM AST

Welcome to your second assignment! This assignment covers our Logic and Proof module. The assignment includes 20 problems that total 95 points. Please submit your work (show what you've done!) as a single PDF on Brightspace. Note that the M, C, and A given in each question represent the marks available for Method, Content, and Answers. Consider these when presenting your work.

Section 1: Logic

These questions apply the techniques we learned towards Logic.

Problem 1 (5 points - 2M/1C/2A): Use the following atomic propositions, translate the following into logical notation:

p : "The password contains at least 8 characters."
q : "The password contains a lowercase letter."
r : "The password contains an uppercase letter."
s : "The password contains a number."
t : "The password contains a symbol (a non-alphanumeric character)."
u : "The password was used in the past year."
v : "The password is valid."

a) The password is not valid if it contains less than 8 characters or if it was used in the past year.

b) To be valid, a password must be at least 8 characters long, not used in the past year, and contain at least two of the following types of characters: symbol, uppercase letter or number.

Problem 2 (5 points - 3M/2C/2A): In addition to logical operations, computer circuitry is built to carry out arithmetic operations very quickly. Behind the scenes, the computer relies on binary numbers. For example, the number $x = 13$ is represented by the binary number 1101_2 in binary. We can denote this binary number as $\langle a_0, a_1, a_2, a_3 \rangle = \langle 1, 0, 1, 1 \rangle$. To see that this is equal to the number

$$\sum_{i=0}^{3} a_i 2^i = 2^0 + 2^2 + 2^3 = 13,$$

i.e. in this binary representation of $13, a_0 = 1, a_1 = 0, a_2 = 1$, and $a_3 = 1$. Each number has exactly one representation. Using only a_0 through a_3, we can represent the numbers 0 (denoted as $\langle 0, 0, 0, 0\rangle$) through 15 (denoted as $\langle 1, 1, 1, 1\rangle$).

In some programming languages, 0 is treated as false and 1 as true (e.g. Python). Thus, if we wanted to check that this number was equal to 13, we could check the truth value of the proposition

$$a_0 \wedge \neg a_1 \wedge a_2 \wedge a_3.$$

Let $x \in \mathbb{Z}$ and $x \in [0, 16)$.

For each of the following conditions, give a compound proposition over the Boolean variables $\{a_0, a_1, a_2, a_3\}$ (think of 0 as false and 1 as true) that expresses the stated condition. To get full marks, your final answer should not be simply a compound proposition of all possibilities.

a) x is divisible by 2

b) x is divisible by 4

Problem 3 (4 points - 1M/1C/2A): Use truth tables to determine if the following logical equivalences hold. Show the truth table and explain why you believe the equivalence does/does not hold:

a) $p \vee (q \wedge r)$ and $(p \vee r) \wedge (q \vee r)$

b) $\neg(p \vee q)$ and $\neg p \vee \neg q$.

Problem 4 (5 points - 1M/2C/2A): Reduce the following statements to the fewest number of literals without using truth tables. That is, use either algebraic manipulation or Karnaugh Maps.

1. $(\sim P \wedge \sim Q \wedge \sim R) \vee (\sim P \wedge Q \wedge R) \vee (\sim P \wedge Q \wedge \sim R) \vee (P \wedge \sim Q \wedge \sim R) \vee (P \wedge \sim Q \wedge R)$
2. $(P \wedge \sim Q \wedge \sim R) \vee (\sim P \wedge \sim R \wedge \sim S) \vee (\sim P \wedge \sim R \wedge S) \vee (\sim P \wedge R \wedge \sim S) \vee (\sim P \wedge Q \wedge R \wedge S) \vee (P \wedge \sim Q \wedge R \wedge \sim S))$

Problem 5 (5 points - 1M/2C/2A): You are working to do some red-eye colour correction on the pixels of an image. The image is represented as a 2-dimensional array holding m rows and n column; i.e. the rows are numbered $1 \ldots m$ and the columns are numbered $1 \ldots n$.

Let $\text{Red}(r, c)$ be a predicate such that for every r and c, $\text{Red}(r, c)$ is true if the pixel in row r column c is red and false otherwise. Translate these statements into predicate logic:

a) There is at least one red pixel.

b) Every row has at least one red pixel.

c) There are never two consecutive red pixels in the same column.

Problem 6 (5 points - 1M/0C/4A): Write out each proposition below in English, and then determine the truth values, where $Q(x, y) : x + y = x - y$. Make sure to show your reasoning/work.

a) $Q(2, 0)$

b) $\forall x \in \mathbb{Z}, \exists y \in \mathbb{Z}, Q(x, y)$.

c) $\forall x \in \mathbb{Z}, \forall y \in \mathbb{Z}, Q(x, y)$.

d) $\exists x \in \mathbb{Z}, \exists y \in \mathbb{Z}, Q(x, y)$.

Problem 7 (9 points - 3M/4C/2A): In this problem, you will write an English sentence that expresses the logical negation of each given sentence. For each:

- Write the initial statement in predicate logic (with your own notation). For example, if the statement was "All squares are rectangles", you would write $\forall x \in S, R(x)$. where S is the set of all squares, and $R(x)$ is the statement "x is a rectangle."
  - If you find the sentence is ambiguous in its meaning, describe the multiple interpretations of the sentence that you can find.
  - Choose one and give its negation in the remaining tasks.
- Write out the negation of the sentence using predicate logic. For our example above, we would have $\exists x \in S, \neg R(x)$.
- Write out the negation of the sentence in plain English. For our example, this would be "there is a square that is not a rectangle."

  Note: Use genuine negation. Starting the sentence with "It is not the case that..." will not count.

a) Every image file is more than 2 megabytes.

b) There exists a number for which no file size can be larger than.

c) Every great programmer plans ahead and comments their code.

Problem 8 (5 points - 1M/2C/2A): Databases store sets of related information. They can be queried to find a subset of information that one is interested in. We will consider (as an approximation) a two-dimension table, where rows hold individual entities and columns correspond to fields (data about those entities).

A database query defines a predicate $Q(x)$ consisting of tests of the values from various columns joined by logical connectives. In response to a query, the database returns the set of entities (all the rows) for which the query $Q(x)$ is true.

Name	**StudentID**	**GPA**	**Major**	**inResidence**	**Home**	...
Angela	B00123456	3.01	CS	False	Halifax	...
Corey	B00654321	2.12	MATH	False	Winnipeg	...
Mohammed	B00321654	3.97	CS	True	Mumbai	...
Ayumi	B00456123	3.81	CS	True	Tokyo	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Student information is stored in such a database (modelled above). To find a list of all CS students with a GPA over $3,5$, we could query

$$[Major(x) = \text{“}CS\text{”}] \wedge [GPA(x) > 3.5].$$

This could return many students, among which it would return Mohammed and Yulin, but not Angela and Corey.

Each of the following predicates $Q(x)$ tests on the specified columns for row x. For each, give a logically equivalent predicate using the rules of logical equivalence given in class in which each column's name appears AT MOST once. Additionally, you may use the symbols True, False, $\wedge, \vee, \neg, \implies$ as many times as you would like.

a) $[(home(x) = \text{“Halifax''}) \wedge (GPA(x) > 3.3)] \wedge \neg[(CS(x)) \vee (GPA(x) \leq 3.3)]$.

b) $inResidence(x) \implies \neg[usesLibrary(x) \implies (inResidence(x) \wedge usesLibrary(x))]$.

Section 2: Direct Proofs

These questions apply the techniques we learned using direct proofs. For each of these, do not forget to include the type of proof you are using, what you assume (the premise), and what you want to show (the conclusion).

Problem 9 (3 points - 1M/1C/1A): Use a direct proof to show that:

If an integer x is odd, then x^3 is odd.

Problem 10 (4 points - 1M/2C/1A): Use a proof by cases to show that:

If two integers have the same parity (both even or both odd), then their sum is even.

Problem 11 (4 points - 1M/2C/1A): Use a proof involving sets to show that:

If $A \subseteq B$ and $B \subseteq C$, then $A \subseteq C$.

Problem 12 (2 points - 1M/0C/1A): Use a proof by example to show that:

There exists a number x such that $x^2 < \sqrt{x}$.

Section 3: Indirect Proofs

These questions apply the techniques we learned using indirect proofs. For each of these, do not forget to include the type of proof you are using, what you assume (the premise), and what you want to show (the conclusion).

Problem 13 (4 points - 2M/1C/1A): Use a proof by contrapositive to show that:

For all integers n, if $5 \nmid n^2$ then $5 \nmid n$, where the symbol $|$ means "divide".

Problem 14 (4 points - 2M/1C/1A): Use a proof by contradiction to show that:

The sum of a rational number and an irrational number is irrational.

Section 4: Inductive Proofs

These questions apply the techniques we learned using inductive proofs. For each of these, do not forget to state and test the base case(s), include your inductive hypothesis, and clearly show when you're using the inductive hypothesis.

Problem 15 (5 points - 1M/3C/1A): Use induction to show that:

For any $n \in \mathbb{N}, n < 2^n$.

Problem 16 (5 points - 1M/3C/1A): Use strong induction to show that:

An object costing \$$n$, $n \geq 8$, can be purchased using only \$3 and \$5 coins.

Section 5: You Pick the Proof

Prove the following statements using the method of your choice. Before you begin your proof, explain why you chose that method. If the proof ends up not working, change methods and explain why you chose the new method. Submit all of your work, including the methods that did not work.

Note: some of these proofs are tricky. Do not fall into the trap of using the result (the conclusion) inside of the proof itself. For instance, for question 12, do not say that $S(n) + c = S(m) + c$ implies $S(n) = S(m)$ since this is what you are trying to show.

Problem 17 (4 points - 2M/1C/1A): If x is a positive irrational number, then $\sqrt{x}$ is also irrational.

Problem 18 (4 points - 2M/1C/1A): If $x, y, z \in \mathbb{Z}$, and the sum $x + y + z$ is odd, then at least one of x, y, and z is odd.

Problem 19 (5 points - 3M/1C/1A): If A, B and C are sets, then $A \cap (B \cap C) = (A \cap B) \cap C$.

Problem 20 (6 points - 2M/3C/1A): Let $n \in \mathbb{N}$. We define the successor of n, denoted by $S(n)$, to be $n + 1$. We assume the following axiom (which is a theorem that we define to be true, so we do not need to prove it):

$$\text{If } S(n) = S(m) \text{ for } n, m \in \mathbb{N}, \text{ then } n = m$$

which essentially says that if we have two natural numbers whose successors are the same (i.e., if we add one to both of them and we get to the same number), then those numbers had to have been equal.

Using this axiom, but **NOT** using subtraction, prove the following:

$$\text{Let } a, b, c \in \mathbb{N}. \text{ Then } a + c = b + c \text{ if and only if } a = b.$$