# CPSC 121: Models of Computation
# Assignment #3

**Due: Friday, October 31, 2014 at 5:00 pm** **Total Marks: 60**

## Submission And Marking Instructions

**All assignments should be done in groups of 2**. It is very important to work with another student and exchange ideas. **Each group should submit ONE assignment**. Type or write your assignment on clean sheets of paper with question numbers prominently labelled. Answers that are difficult to read or locate may lose marks. We recommend working problems on a draft copy then writing a separate final copy to submit.

Your submission must be **STAPLED** and include the **CPSC 121 assignment cover page** – located at the Assignments section of the course web page. Additionally, include your names at the top of each page. We are not responsible for lost pages from unstapled submissions.

Submit your assignment to the appropriately marked box in room **ICCS X235** by the due date and time listed above. **Late submissions are not accepted**.

Note: the number of marks allocated to a question appears in square brackets after the question number.

## A Note on the Marking Scheme

Most items (i.e., question or, for questions divided into parts, part of a question) will be worth 3 marks with the following general marking scheme:

- **3 marks:** correct, complete, legible solution.
- **2 marks**: legible solution contains some errors or is not quite complete but shows a clear grasp of how the concepts and techniques required apply to this problem.
- **1 mark**: legible solution contains errors or is not complete but shows a clear grasp of the concepts and techniques required, although not their application to this problem or the solution is somewhat difficult to read but otherwise correct.
- **0 marks**: the solution contains substantial errors, is far from complete, does not show a clear grasp of the concepts and techniques required, or is illegible.

This marking scheme reflects our intent for you to learn the key concepts and techniques underlying computation, determine where they apply, and apply them correctly to interesting problems. It also reflects a practical fact: we have insufficient time to decipher illegible answers. At the instructor's discretion, some items may be marked on a different scale. TAs may very occasionally award a bonus mark for exceptional answers.

**Question 1 [ 12]**

Consider the following predicates over the set **U** of UBC students:

- ***C(x)***: x is in CPSC 121 class.
- ***J(x)***: x is a junior student
- ***S(x)***: x is a senior student
- ***CS(x)***: x is a CS major
- ***M(x)***: x is a Math major
- ***T(x, y)***: student x tutors student y (i.e. x is a tutor of y).

Rewrite each of the following statements in Predicate Logic using only the given predicates and the operators = and ≠ .

a) [2] Junior students who are in the CPSC 121 class are either CS or math majors.

b) [2] No senior student in the CPSC 121 class is a CS or math major.

c) [2] Every senior math major tutors some junior CS major who is in CPSC 121 class.

d) [2] Some senior CS majors do not tutor any junior student who is in CPSC 121 class.

e) [2] Mary is tutored by at least two senior students who are not in CPSC 121.

f) [2] Every student in CPSC 121 is tutored by exactly one student who is a math major.

**Question 2 [ 12, 2 marks per question]**

Using the definitions of question 1, translate each of the following predicate logic statements into English. Try to make your English translations as natural as possible.

Note: We use the typical order of operations (as the text does): ~ is evaluated first, then ∧ and ∨, and → is evaluated last.

a) $\forall x \in U,\ C(x) \rightarrow \sim\exists\, y \in U, T(x, y) \vee T(y, x)$

b) $\forall x \in U, T(x, x) \rightarrow \sim\exists\, y \in U, (T(x, y) \vee T(y, x)) \wedge x \neq y$

c) $\forall x \in U, \forall y \in U,\ (C(x) \wedge T(y, x)) \rightarrow (\forall z \in U, T(z, x) \rightarrow y = z)$

d) $(\forall x \in U,\ S(x) \rightarrow \exists y \in U, T(x, y)) \wedge (\forall x \in U, J(x) \rightarrow \sim\exists y \in U, T(x, y))$

**e)** $\boldsymbol{\forall x \in U,\ J(x) \rightarrow (\forall y \in U, T(x, y) \rightarrow x \neq y)}$

f) $\forall x \in U, \forall y \in U, \forall w \in U, \forall z \in U, (x \neq y \land T(x, w) \land T(y, z)) \rightarrow (w \neq y \land z \neq x)$

## Question 3 [ 18 ]

Consider again the domain and predicates of question 1with the following additional predicates:

- ***F(x)***: x plays football
- ***N(x)***: x is mean (i.e. a mean person)
- ***MP(x)*** : x likes solving math puzzles

For each of the following arguments translate them into Predicate Logic (using quantifiers and predicates) and provide a formal proof for them.

a) [6]
*Neither CS nor math students have a tutor.*
*Paul is Mary's tutor.*
----------------------------------------------------
*Mary is not a math student.*

b) [6]
*Every CPSC 121 student is a junior student*
*Junior students don't play football*
*Mean students play football*
----------------------------------------------------
*CPSC 121 has no mean students*

c) [6]
*Every student is either a senior or a junior*
*Senior students like solving math puzzles*
*CS students don't like solving math puzzles*
*Charles is a student, but not a junior student*
----------------------------------------------------
*There is at least a student in the university who is not in the CS program*

## Question 4 [ 9 ]

Consider the following theorem:

**If $x$ is a positive integer that is not divisible by $3$, then $x^2 - 1$ is divisible by $3$.**

a) [3] Rewrite the theorems using quantifiers and predicates. You can use predicates for oddness and divisibility, like
   - *Divisible(x,y)* : x is divisible by y

   or for any other numerical properties you need to define.

b) [6] Give a proof for the theorem. Your proof can be informal (in natural language), but it must have a similar structure of a formal proof , with clear steps and correct application of the rules.

Recall that

- to prove a universal statement we start by instantiating the statement with a generic element of the domain and prove it for that item
- to prove a existential statement we need to find an element of the domain and prove that the statement is true for that element.

*Hint for this proof*: if an integer is not divisible by `3`, then it can be written as either `3k+1` or as `3k+2` for some integer `k`.

## Question 5 [ 9 ]

Suppose that an algorithm A executes $5n^3 + 20n^2 + 65n + 200$ steps when run with input of size n. Prove that the algorithm is in $O(n^3)$ (i.e. A is in big-O of $n^3$) .

a) [3] Using the predicate we defined in the class for the big-O notation, translate this statement into predicate logic (that is, using predicates and quantifiers).

b) [6] Give a proof for this statement. Your proof can be a formal proof or informal (in natural language), but it must have a similar structure of a formal proof.