HAND IN
Answers recorded on exam paper

QUEEN'S UNIVERSITY FINAL EXAMINATION

FACULTY OF ARTS AND SCIENCE

School of Computing

CISC/CMPE 422 – Professor Nicolas Hili

December 9, 2017

INSTRUCTIONS TO STUDENTS:

This examination is 3 HOURS in length.

There are five questions to this examination, some with several parts.

Please answer all questions in the exam.

The following aids are allowed:
One 8.5"x11" data sheet with information on both sides.

GOOD LUCK!

PLEASE NOTE:

Proctors are unable to respond to queries about the interpretation of exam questions.

Do your best to answer exam questions as written.

Question 1: Propositional & Predicate Logics (15 points)

Id: ____________________

Consider the following two predicate logic formulas

$$\varphi_1 = \forall x.(p(x) \to \neg\exists y.q(f(x,y)))$$
$$\varphi_2 = \forall x.\neg\exists y.(p(x) \to q(f(x,y)))$$

where x and y are variables, p and q are predicate symbols of arity 1, and f is a function symbol of arity 2.

φ_1 and φ_2 are not equivalent. More precisely, φ_1 does not imply φ_2. Find a model $\mathcal{M}$, such φ_1 holds in $\mathcal{M}$, but φ_2 does not hold in $\mathcal{M}$. Make sure you provide a complete definition of your model $\mathcal{M}$.

Your answer:

Question 2: Alloy (32 points)

Id: ______________________

To support the enrollment of students to courses offered in a specific program, the following Alloy model M is constructed:

```
sig Course {
  studentsEnrolled : set Student,
  prerequisites : set Course
} {
  one this.~courses
}

sig Program {
  courses : some Course
}

sig Student {
  programFollowed: Program,
  coursesPassed: set Course,
  coursesEnrolled: set Course
}

fact { studentsEnrolled = ~coursesEnrolled }

pred Show {
  some Program
  some Student
}
run Show for 3
```

The *studentsEnrolled* attribute associates a set of students enrolled with a specific course. The inverse attribute of *studentsEnrolled* is *coursesEnrolled*. The *coursesPassed* attribute associates a set of courses that have been passed with the students who passed them (that is, according to Queen's policy, students who got a D- mark or above). The *prerequisites* attribute associates with each course a set of prerequisite courses.

a) (4 points) Draw a consistent object model, i.e., a model that satisfies the specification (using the predicate *Show*).

Your answer:

Question 2: Alloy (continued) Id: ____________

b) (4 points) Draw the graphical representation of M (i.e., the "metamodel" or "class diagram" corresponding to M). Make sure you also include any multiplicity constraints expressed in M. You can use either the Alloy or the UML notation (but not both).

Your answer:

c) (4 points) Define a *prereq* function that for a course c returns all its prerequisite courses (i.e., the direct and indirect prerequisite courses)

Your answer:

```
fun prereq[c : Course] : set Course {

}
```

For questions d) to h), using the Allow specification above, express each of the following invariants formally in Alloy. You can reuse the *prereq* function defined above.

d) (4 points) *"The prerequisite relationship does not contain any cycles"*

Your answer:

Question 2: Alloy (continued) Id: ____________

e) (4 points) *"A student following a specific program cannot be enrolled in courses that are not part of that program"*

Your answer:

f) (4 points) *"A course of a specific program cannot have prerequisite courses from another program"*

Your answer:

g) (4 points) *"A student cannot be enrolled in a course he or she has already passed"*

Your answer:

h) (4 points) *"A student cannot be enrolled in a course if he or she does not have the prerequisite courses"*

Your answer:

Question 3: CTL (18 points)

Id: ____________________

Consider the three pairs of non-equivalent formulas φ_i, φ'_i below. For each pair, find a Kripke Structure that distinguishes them. More precisely, for each pair, draw a Kripke Structure M_i such that one formula holds in M_i, but not the other.
Important: When drawing M_i, make sure that you clearly indicate (1) the initial state of M_i, (2) which atomic propositions occuring φ_i and φ'_i hold in which states of M_i, and (3) which of the two formulas holds in M_i. Also, remember that the transition relation of a Kripke Structure is total.

a) (6 points) $\varphi_1 = \neg\mathbf{AG}\, p$ and $\varphi'_1 = \mathbf{AG}\,\mathbf{AX}\,\neg p$

Your answer:

b) (6 points) $\varphi_2 = \mathbf{A}[p\ \mathbf{U}\ \mathbf{EG}\ q]$ and $\varphi'_2 = \mathbf{A}[p\ \mathbf{U}\ q]$

Your answer:

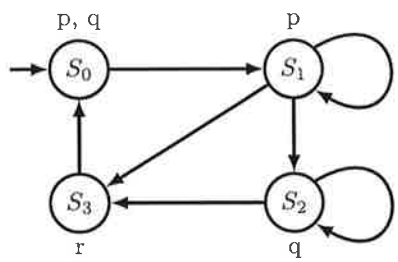
c) (6 points) $\varphi_3 = \mathbf{EG}\, p$ and $\varphi'_3 = \mathbf{EG}\,\mathbf{AG}\, p$

Your answer:

Question 4: Model checking (26 points)

Id: ____________________

Consider the following graphical representation of a Kripke Structure (i.e., finite state machine) M.



a) (4 points) Define M textually as a 4-tuple $M = (S, S_0, R, L)$. Make sure you define all components of M.

Your answer:

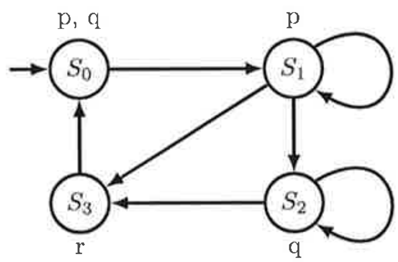
b) (3 points) For each of the following statements, indicate whether the statement is true or false by circling the corresponding answer.

Your answer:

From every state, all states are reachable:	true	false
The relation R is total:	true	false
The Kripke structure is deterministic:	true	false

Question 4: Model checking (continued) Id: ______________

For your convenience, the Kripke Structure M from the previous page is repeated here.



c) (12 points) In the table below, the rows are labeled with CTL formulas φ_i and the columns are labeled with states s_j of M. For each empty cell (φ_i, s_j) in the table, determine whether or not M in state s_j satisfies formula φ_i, i.e., if $(M, s_j) \models \varphi_i$ or $(M, s_j) \not\models \varphi_i$. Write *"Yes"* into cell (φ_i, s_j), if $(M, s_j) \models \varphi_i$. Write *"No"*, otherwise. For instance, **AX** $(p \vee q)$ is satisfied in state s_0, but not satisfied in state s_1, i.e., $(M, s_0) \models$ **AX** $(p \vee q)$ and $(M, s_2) \not\models$ **AX** $(p \vee q)$.

	s_0	s_1	s_2	s_3
AX $(p \vee q)$	Yes	No	No	Yes
A$[p$ **U** $r]$				
AG EX $(q \rightarrow r)$				
EG EF $(p \wedge q)$				

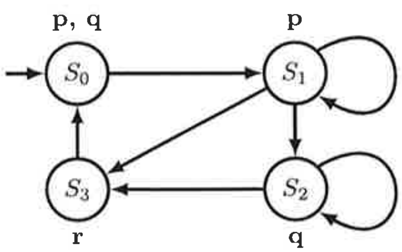
d) (7 points) The formula **E**$[\ \psi_1$ **U** $\psi_2]$ (where $\psi_1 = p \vee r$ and $\psi_2 = \neg$**EX** p) holds for the initial state s_0. Demonstrate this on the next page by handsimulating the CTL model checking algorithm on the Kripke structure M. Use the copies of M on the next page to clearly show how the info about which (sub)formula holds in which states is propagated through M (as seen in class). Use as many steps as you want. Treat the different connectives explicitly (i.e., do not translate the formula into another formula only using an adequate set of connectives).

Hint: there should be no less than 3 steps and no more than 8 steps.

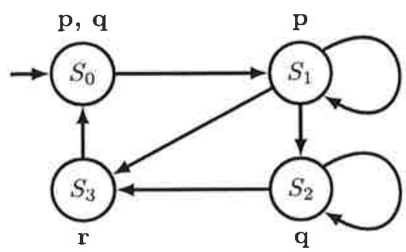
Question 4: Model checking (continued)

Id: ____________________

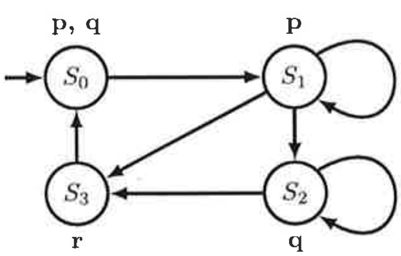
Step 1



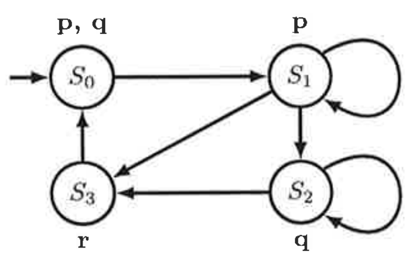
Step 2



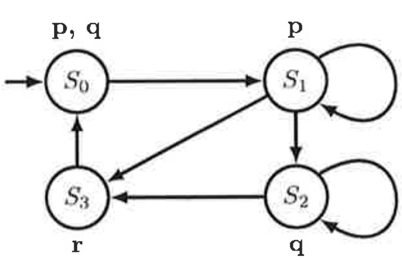
Step 3



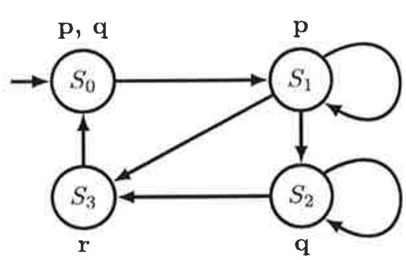
Step 4



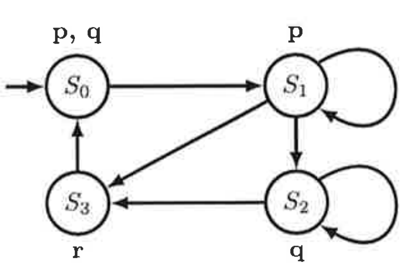
Step 5



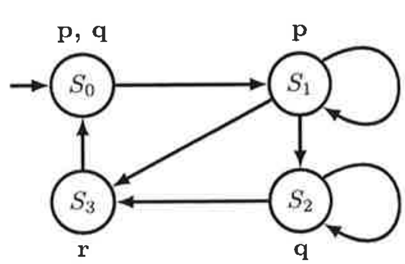
Step 6



Step 7



Step 8



Question 5: SMV (35 points)

Id: ____________________

Consider the following SMV code.

```
MODULE main

VAR
  x : {1, 2, 3};
  y : {a, b};
  z : boolean;

ASSIGN
  init(x) := 1;
  next(x) := case
               x=1 : {1, 2};
               x=2 : 3;
               x=3 : 1;
             esac;

  init(y) := a;
  next(y) := case
               (x=1 | x=3) : a;
               TRUE : b;
             esac;

  init(z) := FALSE;
  next(z) := case
               y=b : !z;
               TRUE : x;
             esac;
```

a) (6 points) Let M be the Kripke structure defined by the SMV code above. We represent a single state s of M by a triple (x, y, z) where x is the value of variable x, y is the value of variable y, and z is the value of variable z. For instance, the initial state of M is represented as $(1, a, \mathit{false})$.

Beginning from the initial state of the machine M, draw the computation tree T of M. Draw T to a depth of 4. More precisely, you must show all computation paths of M starting at the initial state up to length 4, where the length of a path is the number of edges on it.

Your answer:

 Id: ____________________

For your convenience, the SMV code from the previous page is repeated here.

```
MODULE main

VAR
  x : {1, 2, 3};
  y : {a, b};
  z : boolean;

ASSIGN
  init(x) := 1;
  next(x) := case
               x=1 : {1, 2};
               x=2 : 3;
               x=3 : 1;
             esac;

  init(y) := a;
  next(y) := case
               (x=1 | x=3) : a;
               TRUE : b;
             esac;

  init(z) := FALSE;
  next(z) := case
               y=b : !z;
               TRUE : x;
             esac;
```

b) (4 points) Using computation tree T from the previous question as a guide, draw the Kripke structure M that is defined by the SMV code above. Your drawing should clearly indicate the initial states of M, the reachable states of M, and the transition relation of M. You don't need to show the labelling function.

Your answer:

Question 5: SMV (continued)

Id: ____________________

c) (25 points) For each of the following properties φ_1 through φ_5, express it formally in CTL and decide whether or not it is true in your finite state machine M given in Part 1 of this question by writing **"Yes"** or **"No"**. No justification necessary.

1. (5 points) φ_1: "Along every path it is always the case that if x is 1 or 2, then y is equal to a".

Your answer: φ_1 in CTL:
φ_1 true in M?:

2. (5 points) φ_2: "For all states s along every path, if x is equal to 1 in s, then s has a successor in which x is 1 and another successor in which x is 2".

Your answer: φ_2 in CTL:
φ_2 true in M?:

3. (5 points) φ_3: "Along every path it is always the case that x is 3 if and only if y is equal to b".

Your answer: φ_3 in CTL:
φ_3 true in M?:

4. (5 points) φ_4: "There exists a path along which x is always equal 1, y is always equal to a, and z never holds".

Your answer: φ_4 in CTL:
φ_4 true in M?:

5. (5 points) φ_5: "Along every path and in every state, we always eventually reach a state in which x equal to 1 and z doesn't hold".

Your answer: φ_5 in CTL:
φ_5 true in M?:

Scratch sheet: **Id:** ____________________