Final

Note

This is a take-home exam. You may consult any books, papers, or notes, but discussion with other students or seeking outside help is strictly forbidden. Please **use official NTU exam paper** to write up your solutions. Drop your solutions in Yih-Kuen Tsay's mail box or email him an electronic version by the deadline.

Problems

1. (40 points) This follows up Problem 5 of HW#1. Please use the direct and the reverse simulations described in Section 5 of [1] to optimize the generalized Büchi automaton obtained from $\mathbf{G}(p \rightarrow \mathbf{F}q)$ (which should have five states, with labels on the states). Keep the labels on the states. Describe the simulation relations you found and show they indeed are simulations. Please show the calculation for the simulation found. As you will see, several states may be removed. Can you explain, using the information carried on the states (i.e., the sub-formulae), why such simplification is possible for this particular case?

2. (30 points) The following is a NuSMV module containing two asynchronous processes.

```
MODULE main
VAR
  semaphore : boolean;
  proc1     : process user(semaphore);
  proc2     : process user(semaphore);
ASSIGN
  init(semaphore) := 0;

MODULE user(semaphore)
VAR
  state : {idle, entering, critical, exiting};
ASSIGN
  init(state) := idle;
  next(state) :=
    case
      state = idle                  : {idle, entering};
      state = entering & !semaphore : critical;
      state = critical              : {critical, exiting};
```

```
      state = exiting               : idle;
      1                             : state;
    esac;
  next(semaphore) :=
    case
      state = entering : 1;
      state = exiting  : 0;
      1                : semaphore;
    esac;
```

(a) Write a boolean formula that characterizes all reachable states of this module.

(b) Please draw a BDD (as small as possible) for the formula in 2a.

3. (30 points) Find a regular language which the L^* algorithm needs exactly two conjecture queries to learn. Please detail the L^* learning process of the language you found. Can you give general rules or heuristics via which one can quickly tell that more than one conjecture queries will be needed?

References

[1] F. Somenzi and R. Bloem. Efficient Büchi automata from LTL formulae. In *Proceedings of the 12th International Conference on Computer-Aided Verification (CAV 2000), LNCS 1855*, pages 248–263. Springer, 2000.