CS_M13
CRITICAL SYSTEMS
Exam January 2014

(*Attempt* **2** *questions out of 3*)

Question 1. *(Terminology, safety criteria, verification, hazard analysis)*

(a) Define briefly, what a critical system is. Which aspects of a web browser could be considered as a critical system (even though they are usually not treated as such)?

[3 marks]

(b) The control system of a space probe is (if used only when in space) unlikely to endanger the lives of human beings. Explain why it is still considered as a critical system.

[3 marks]

(c) Assume two versions of a telephone exchange server. Version A fails on average once per calendar month (30 days) for 5 seconds, version B fails once per year (for simplicity considered as 360 days) for 50 seconds. Determine the availability of both systems. It suffices to write down your result as a formula. Which of the two systems is more available and which is more reliable? Explain your result.

[8 marks]

(d) The company RailCorrect has developed a new railway control system, the correctness of which has been fully verified using automated and interactive theorem proving techniques. When they start testing the system, it turns out that from time to time trains seem to vanish from the screen. Since the system is fully verified, it is clear that the software fulfils the specification. Which part of the development process for this software went wrong?

[4 marks]

(e) Give a fault tree analysis of a system which will trigger an alarm in case the content of a tank containing a dangerous liquid reaches a too high level. You can assume that the system consists of two independent subsystems, each consisting of a sensor activated when the liquid reaches a too high level and a siren. Start with the failure of both subsystems. Consider as causes for the failure that the siren is not working or the sensor is not working. Your fault tree analysis should contain at least one or-gate and at least one and-gate.

[7 marks]

Please turn over for Question 2

Question 2. *(Race conditions, Programming languages for critical systems)*

(a) Give 3 reasons which make the full language of Ada **not** suitable for critical systems.

[6 marks]

(b) When developing SPARK Ada, it was decided to use a subset of Ada rather than developing a new language. Determine 2 reasons why this is an appropriate decision.

[6 marks]

(c) One requirement for programming languages for safety critical applications is verifiability. What is meant by this? Give 2 features of SPARK Ada which support verifiability.

[5 marks]

(d) Explain what is meant by a race condition. Race conditions are difficult to detect through testing. Why? Consider the scenario of a system for allocating ambulances, consisting of one central unit administrating the allocation of ambulances to emergencies, and several units making requests for ambulances concurrently. Describe a scenario, in which, due to bad implementation, race conditions could result in a request for an ambulance being acknowledged, but the ambulance being allocated for a different emergency case.

[8 marks]

Please turn over for Question 3

Question 3. *(SPARK Ada)*

(a) Which Ada compilers can be used for compiling SPARK Ada code to obtain executable code, and which not?

[3 marks]

Part (b) - (e) are related to the following incomplete procedure written in SPARK Ada, which tries to transfers the amount TransferAmount from bank account Account1 to bank account Account2:

```
procedure Transfer(Account1       : ?? Integer;
                   Account2       : ?? Integer;
                   TransferAmount : ?? Integer;
                   Success        : ?? Boolean)
--# derives ?? ;
--# pre  Account1 >= 0 and Account2 >= 0;
--# post Account1 >= 0 and Account2 >= 0;

is
begin
  if (Account1 - TransferAmount >= 0)
  then
    Account1 := Account1 - TransferAmount;
    Account2 := Account2 + TransferAmount;
    Success  := True;
  else
    Success  := False;
  end if;
end Transfer;
```

(b) Complete the signature of the procedure Transfer such that it will pass SPARK Ada's data flow analysis. The procedure should not make use of any global variables.

[5 marks]

(c) Complete the derive clauses, so that the program passes SPARK Ada's information flow analysis.

[6 marks]

(d) Derive verification conditions from the program. You can ignore any conditions regarding that integers need to be between a minimum and maximum integer.

[6 marks]

(e) The verification conditions will not be provable in this case. Why? Modify the pre or post conditions so that those conditions become provable.

[5 marks]