Concordia University

Department of Computer Science and Software Engineering

SOEN 331-S:

Formal Methods for Software Engineering

Assignment 3

Dr. Constantinos Constantinides, P.Eng.

constantinos.constantinides@concordia.ca

November 3, 2022

Contents

1 General information **3**

2 Introduction **3**

3 Ground rules **3**

4 System specification **4**

4.1 Top level specification 4

4.2 Active 4

4.3 Reading cycle 5

4.4 Emergency 6

5 Your assignment **6**

6 What to submit **7**

1 General information

Date posted: Thursday 3 November, 2022.

Date due: Monday, 21 November, 2022, by 23:59. (REVISED)

Weight: 10% of the overall grade.

2 Introduction

You should find one partner and between the two of you should designate a team leader who will submit the assignment electronically.

3 Ground rules

This is an assessment exercise. You may not seek any assistance while expecting to receive credit. **You must work strictly within your team and seek no assistance for this assignment (e.g. from the teaching assistants, fellow classmates and other teams or external help)**. You should **not** discuss the assignment during tutorials. I am available to discuss clarifications in case you need any.

Both partners are expected to work relatively equally on each problem. Accommodating a partner who did not contribute will result in a penalty to both. You cannot give a "free pass" to your partner, with the promise that they will make up by putting more effort in a later assignment.

If there is any problem in the team (such as lack of contribution, etc.), the team leader must contact me as soon as the problem appears.

4 System specification

Consider an alarm system that monitors (a) temperature and (b) carbon monoxide (CO) levels. Carbon monoxide measurement indicates the presence of CO in *parts per million* (ppm).

4.1 Top level specification

Initially the system would be idle. While idle, the system can be activated, or shut off. While active, the system can become idle but it cannot be shut off. The system can only be shut off while idle.

4.2 Active

Once activated the system enters some initial mode whereby it displays 'Configuring mode' on some screen and allows its user to perform some basic configuration by setting the thresholds for temperature and CO level (above either of which the alarm would go off). The system acknowledges the setting of each threshold with a double beep. If the user attempts to confirm the setting of any of the two thresholds with an illegal value, then the system will reject this setting and will generate an error. An illegal value for a temperature threshold would be one that is not greater than the current room temperature. An illegal value of the CO level threshold would be one that is not between 100 and 120. In other words, the system provides little flexibility on the threshold range for CO.

The system can allow its user to skip configuration if this has already been set and no additional settings are required.

Upon entering and confirming proper threshold settings, the system moves to a reading mode whereby it will read in the current temperature level and the current CO level in a series of cycles. While in reading mode, the system continuously slowly blinks some red led. The system will initially indicate that has entered the reading mode by getting into some idle mode whereby it performs a sequence of beeps. After 5 seconds it will leave this idle mode

(generating a prolonged beep to indicate so), and it will enter a monitoring mode that serves as the *starting point* of the cycle of temperature and CO level readings.

4.3 Reading cycle

The system stays in this starting point for 15 seconds and then it proceeds to read in the current temperature and the current CO level.

- **Reading temperature**: If the current temperature is between 10 and 5 degrees (inclusive) below its set threshold then the system sets on an orange led light and it will proceed to read the current CO level. If the current temperature is less than 5 degrees below its set threshold then the system lights on some red led and it will proceed to read the current CO level. If the temperature measurement falls outside of these ranges from below (i.e. measurement is less than 10 degrees below the threshold), then the system proceeds to read in the current CO level.

- **Reading CO level**: If the current CO level is between 50 (inclusive) but less then 75, then the system sets on an orange led light (provided it is currently off) and it will proceed to the starting point. If the current CO level is between 75 (inclusive) and less than the CO threshold, then the system lights on some red led (provided it is currently off) and it will return to its starting point. If the CO level falls outside of these ranges from below (i.e. the measurement is less than 50), then the system returns to its starting point.

The description above completes one *reading cycle*. The system will repeat this cycle in 15 seconds as indicated above. From the starting point and before the reading of the two values, the system switches off any leds that may be on.

While in reading mode, the system can allow the user to enter configuration mode and re-set the thresholds. As the system exits the reading mode, any leds that may be on are switched off.

4.4 Emergency

If while at reading mode the current temperature or the current CO level reach (or exceed) their corresponding threshold levels, then the system enters an emergency mode while at the same time it sends some notification. We may assume that this notification is sent to some external system (like e.g. a security provider or the Fire Department).

While in emergency mode the system produces a siren sound. The system can exit the emergency mode through a reset button at which point it will display 'Exit emergency' on its display. Upon exiting the emergency mode, the system goes back into a reading mode.

5 Your assignment

1. (80 points) Create the state transition diagram of a UML state machine that models the system above. Use any drawing tool

2. (20 points) Translate the UML state machine into a declarative model (Prolog database). Make sure you maintain a record of all your interaction with the Prolog interpreter and execute the following queries:

   (a) What events, if any, are legal while the system is at any given mode, e.g. while being active? Create a rule that succeeds by encapsulating any such events into a list, and returns the list.

   (b) Create a rule that succeeds by collecting all system actions into a list and returns such list.

   (c) Create a rule that for a given pair of `source-destination` states, the rule succeeds by returning a list of criteria under which the system can perform a transition from the source to the destination. The criteria are defined as `event-guard` pairs.

   (d) Create a rule that succeeds by returning a list of `source-destination` state pairs, not taking into consideration any recursive or internal transitions.

6 What to submit

You must produce a single `pdf` file with any and all state transition diagrams that model the system specification as a UML state machine. You must also provide two additional files: A `pl` file that contains your declarative model and any and all rules, and a `txt` file contains your interaction with the Prolog interpreter according to the queries defined above. Please package all files into a single `zip` file. Name the `zip` file after the Concordia id of the person who will submit, e.g. `123456.zip`, and submit it at the Electronic Assignment Submission portal at

(`https://fis.encs.concordia.ca/eas`)

under **Assignment 3**.

END OF ASSIGNMENT.