UNITED INTERNATIONAL UNIVERSITY (UIU)

Dept. of Computer Science & Engineering

Trimester: Summer 2022

Course No: CSE 4495 Title: Software Quality Assurance and Testing

Section: A

Assignment - 2

There is a single problem in this assignment. Discuss with your teammates and submit one file (per group) in eLMS containing the solution. No more than 5 students per group is allowed.

Cover Page: On the cover page of your assignment, include the name of the course, the date, your group name, and a list of your group members.

Problem Description:

In your previous class test you were introduced to a simple microwave controller modeled as a finite state model. Let's revisit the states of the model-

- **Door: {Open, Closed}** -- sensor input indicating state of the door
- **Button: {None, Start, Stop}** -- button press (assumes at most one at a time)
- **Timer: 0...999** -- (remaining) seconds to cook
- **Cooking: Boolean** -- state of the heating element

Now in this assignment your job is threefold-

1. Designing the finite state model and implementing it in **NuSMV.**
2. Writing a list (at least five) of informal requirements for this microwave controller and expressing them in temporal logic (CTL or LTL). You can take help from your class test question ;)
3. Actually checking these if these requirements hold against your implemented **NuSMV** model.

So your report will have three sections –

Section 1 : NuSMV implementation (8 points):

In this section first design the finite state model (FSM). Next code up the model in NuSMV notation. Your code will look something like this –

```
MODULE
     microwave
VAR
     Door: {Open, Closed};
     Button: {None, Start, Stop};
     Timer: 0..999;
     Cooking: boolean;
```

```
ASSIGN
    init(Door) := Closed;
    init(Button) := None;
    init(Timer) := 0;
    next(Timer) := case
        Timer > 0 & Cooking=TRUE : Timer - 1;
        Timer > 0 & Cooking=FALSE & Button!=Stop : Timer;
        Button=Stop : 0; Timer=0 : 0..999;
        TRUE: Timer;
    esac;

    init(Cooking) := FALSE;
    next(Cooking) := case
          --Write your code here to complete
          -- Suggestion: Start by defining the
          -- conditions that would cause
          -- cooking to start. Then add conditions
          -- that would make it stop.
          -- Finally, ensure it will continue
          -- running if it is supposed to. (FILL THIS IN);
     esac;
```

As you can see half of the code (change of Timer variable) is already done for you. Finish the other half and include the complete code in your report. You should create a **.smv** file with this code.

Section 2 : Requirement Generation (5 points):

- Write down at least 5(five) informal requirements for your model.
- Translate them into temporal logic expression (minimum 3 CTL and 2 LTL expressions).
- Include these requirements in your .smv file created in section-1

Section 3 : Finite State Verification (7 points):

Now verify your model against these requirements. To complete this section visit the NuSMV website: https://nusmv.fbk.eu/. Download and run NuSMV 2.6.
Execute your .smv file with NuSMV and add the consequent screenshot of the NuSMV Console to your report. After you finish this section you have completed your assignment. Congratulations!

Submission Instructions & Deadline:

Now put your .pdf report and .smv source code file inside a folder and zip the folder**. Submit the zipped folder in eLMS window. Only the team leader's submission is sufficient.** Remember to mention all teammates' name and IDs' on the report cover page.

Deadline : 27-09-2022 (Tuesday) 11:59pm. This deadline is non-negotiable and no submissions will be accepted post deadline, regardless of any unfortunate circumstances.