CS3642-01 Programming Assignment #1 (Fall 2023)

Due: September 16, 2023 (11:30 PM)

To implement a model-based AI agent to carry out a simple task for you. Remember that we have learned four different types of AI Agents in Chapter 2. Please show your design and what tasks you try to solve by using the agent you implemented. **You must write your own codes for the algorithms. Make sure your submission meets all of the requirements and free of plagiarism.**

You may write your code in a contemporary language of your choice; typical languages would include C/C++, Python, Java, Ada, Pascal, Smalltalk, Lisp, and Prolog. A GUI interface is preferred.

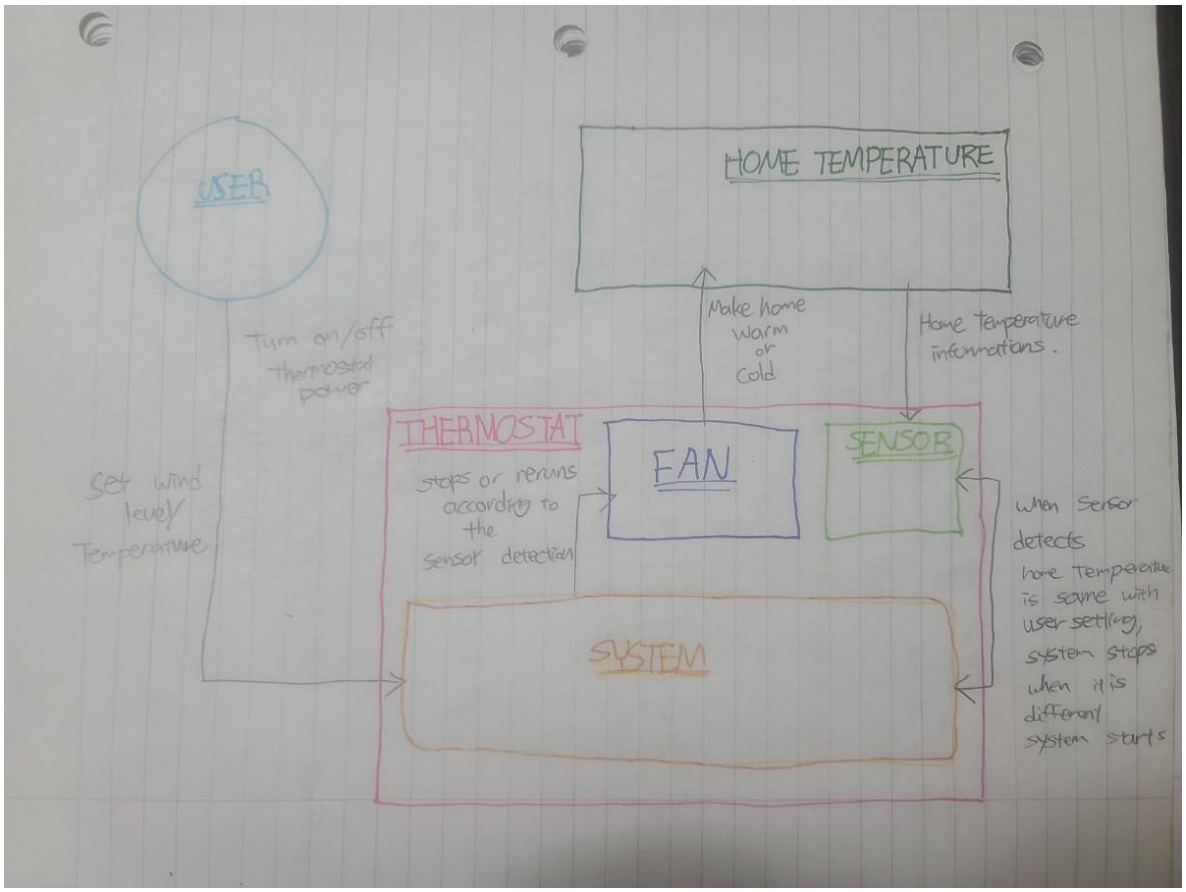
1. Submit a PDF file of your well-commented source program, your design and your printed outputs (screen shots). Please include your codes in your PDF file. It is plagiarism to take any codes from the website or others. Try to understand the algorithm and implement the algorithm by your own. You must have the following 3 sections in your PDF file.
2. Provide a video presentation of the assignment in MP3 or YouTube.
3. Please upload items 1) and 2) separately to D2L. Restriction: No zipped files.

Adding the following 3 sections (I, II, and III) at the beginning of your PDF including your code and outputs.

I. Your information:

// Course: Artificial Intelligence
// Student name: Raehyeong Lee
// Student ID: 000996758
// Assignment #: Assignment 1
// Due Date: 2023 September 16, 2023
// Signature: Raehyeong Lee (Your signature assures that everything is your own work. Required)
// Score: _____(Note: Score will be posted on D2L)

II. Design of your Agent:



III. Tasks that your agent will solve:

Code of Agent

```
import java.util.Scanner;

public class Thermostat{
        private boolean power;
        private boolean system;
        private int setTemp;
        private int currTemp;
        private boolean cooler;
        private boolean heater;
        private int windLevel;

        public Thermostat() {
                Scanner scan = new Scanner(System.in);
                System.out.print("Set Actual Temperature of your room: ");
                currTemp = scan.nextInt();
                power = false;
                system = false;
                setTemp = 500;
                cooler = false;
                heater = false;
                windLevel = 1;
        }
```

```
        public void thermostatInfo() {
                String PonOffStr = (power) ? "ON" : "OFF";
                String SonOffStr = (system) ? "ON" : "OFF";
                final String[] windStr = {"", "level 1", "level 2", "level 3"};

                System.out.println("\n===============Thermostat
Status================");
                System.out.println("Power status:  " + PonOffStr);
                System.out.println("System status: " + SonOffStr);
                System.out.println("Current Temperature (in degree Celsius): " +
currTemp);
                System.out.println("Temperature Setting(in degree Celsius):  " +
setTemp);
                System.out.println("Wind Level: " + windStr[windLevel]);
        }

        public void setThermostatState(){
                Scanner scan = new Scanner(System.in);
                System.out.println("Current Temperature is " + currTemp + ".");
                System.out.print("Set Preferred Temperature: ");
                setTemp = scan.nextInt();
                System.out.print("Set Wind Level: ");
                windLevel = scan.nextInt();

                if (currTemp == setTemp){ // Reached preferred Temperature
                        system = false;
                        power = true;
                        System.out.println("Thermostat stops.");
                        cooler = false;
                        heater = false;

                }else if (currTemp > setTemp) { //HOT
                        system = true;
                        power = true;
                        System.out.println("Cooling system activated.");
                        cooler = true;
                        heater = false;
                } else if (currTemp < setTemp) { //Cold
                        system = true;
                        power = true;
                        System.out.println("Heating system activated.");
                        heater = true;
                        cooler = false;
                }
        }
        public void setSystem(boolean SOnOff) {
                system = SOnOff;
        }
        public void setPower(boolean POnOff) {
                power = POnOff;
        }
        public boolean isHeater() {
                return heater;
        }
        public boolean isCooler() {
                return cooler;
```

```
        }
        public void setCurrTemp(int n) {
                currTemp += n;
        }
        public int getCurrTemp() {
                return currTemp;
        }
        public int getSetTemp() {
                return setTemp;
        }
        public int getWindLevel() {
                return windLevel;
        }
}
import java.util.Scanner;
import static java.lang.Thread.sleep;

public class ThermostatMain {
    public static void main(String[] args) throws InterruptedException {
        Scanner scan = new Scanner(System.in);
        Thermostat Thermostat = new Thermostat();
        int elapsedTime = 0;
        boolean SetTemp = false;
        boolean SetPower = true;
        System.out.print("Turn the Thermostat On?[yes/no]: ");
        String PowerOn = scan.next();

        if(PowerOn.equalsIgnoreCase("YES")) {
            Thermostat.setSystem(true);
            Thermostat.setPower(true);
            Thermostat.setThermostatState();
            while(SetPower) {
                sleep(1000);
                elapsedTime++;
                switch(Thermostat.getWindLevel()) {
                    case 1 :
                        if(elapsedTime >= 5) SetTemp = true; break;
                    case 2 :
                        if(elapsedTime >= 3) SetTemp = true; break;
                    case 3 :
                        if(elapsedTime >= 1) SetTemp = true; break;
                    default:
                }
                if (SetTemp) {
                    if (Thermostat.getCurrTemp() == Thermostat.getSetTemp()) {
                        Thermostat.setCurrTemp(0);
                        Thermostat.setSystem(false);
                        System.out.print("Do you want to turn the Thermostat Power
OFF?[yes/no]: ");
                        String PowerOFF = scan.next();
                        if (PowerOFF.equalsIgnoreCase("YES")) {
                            SetPower = false;
                            break;
                        }else {
                            SetPower = true;
                        }
                    }else if (Thermostat.isHeater()) {
```

```
                        Thermostat.setCurrTemp(1);
                        Thermostat.setSystem(true);
                        Thermostat.setPower(true);
                    }else if (Thermostat.isCooler()) {
                        Thermostat.setCurrTemp(-1);
                        Thermostat.setSystem(true);
                        Thermostat.setPower(true);
                    }
                    Thermostat.thermostatInfo();
                    SetTemp = false;
                    elapsedTime = 0;
                }
            }
        }
    }
}
```

Output Screenshot