AERO 552: Aerospace Information Systems
Homework 3

Out Wednesday, October 11th, 2023

Due on CANVAS Friday, October 20th, 2023, **11:59pm**

Instructions

Your solution should be submitted as

- a file `hw3.pdf` containing your solution to the handwritten problems;
- files `turing.h` and `turing.cpp` containing your solution to Exercise 2, as well as a number of test cases in `.dat` files.

High-level discussion of problems and directions with other students is fine, but exchanging solutions or code is not. If in doubt, please ask. **If you discussed with another student, please indicate so explicitly in your homework, with their name.**

Exercise 1 – Moore Machine: a simplified elevator (20 points)

Define a Moore machine control system for an elevator in a 3-story building. There are 3 buttons the user can press in the simplified elevator: 1, 2, and 3. These numbered buttons allow the user to indicate the next floor, following the conventional interpretation. For the simplified elevator it is fine to ignore button presses that occur between the time the first button is pressed and when the elevator door opens at the chosen floor. The following actions are available to the elevator: UP (travel up one floor), DOWN (travel down one floor), STOP (cease moving up or down), OPEN (open the door), CLOSE (close the door), and NOOP (do nothing). The behavior of the elevator is as follows. The elevator starts on floor 1 with the door open. When the user presses a button that requires transit to a new floor, the elevator door closes, the elevator moves up (or down), and stops when reaching the desired floor. The elevator door then opens and remains open at the current floor until another button is pressed. If the user presses the button corresponding to the current floor the elevator does not move or close its door.

Exercise 2 – Binary Decision Diagrams (20 points)

Construct BDDs for each of the following logic expressions. Order your variable tests so that branching is minimized for each BDD (multiple minimum-branching orderings may exist). Note that you may benefit by simplifying the logical expressions using principles of Boolean Algebra before constructing each BDD. Note that the following notation is used: $\wedge$ represents "and", $\vee$ represents "or", $\neg$ represents "not", and $x \rightarrow y$ represents "if x then y". Also note that $\wedge$ binds tighter than $\vee$.

- $(\neg a) \wedge (a \vee b) \vee (b \vee a) \wedge (a \vee \neg b)$
- $(a \vee b \wedge c) \rightarrow d$

Exercise 3 – Turing Machine simulator (30 points)

Take your implementation of an automaton simulator from homework 1, and extend it to implement a Turing Machine simulator. Write a function `string turing(string file, string input)` that implements this Turing Machine. The function should return string `"REJECT"` if ending up in the reject state, otherwise it should return what's on the tape in the end if ending up in the accept state. To avoid confusions with blank spaces and special characters (while staying within ASCII text files), we will write `w` for the blank space and `E` for the left end-marker. Use a similar format as in Homework 1, following the example:

```
-- alphabet
a b
-- tape alphabet
w E c d
-- states
1 2 3
-- initial
1
-- accept
2
-- reject
3
-- transitions
1 -- a,b,L --> 2
1 -- b,a,L --> 1
2 -- a,b,R --> 3
2 -- b,a,R --> 2
3 -- a,b,L --> 3
3 -- b,a,R --> 3
```

Transition `1 -- a,b,L --> 2` means that on reading `a` in state 1, the automaton should write `b` move the tape-reading head to the left (`L`) and transition to state 2. Note that under `tape alphabet` I have only entered the characters not already in the `alphabet` (to avoid duplication), but of course the tape alphabet also contains the input alphabet.

Exercise 4 – Turing Machine: square function (25 points)

Give a Turing machine with input alphabet $\{a, b\}$ that on input a^m accepts with a^{m^2} written on its tape. The Turing machine should reject any input containing any b. Describe the operation of the machine both informally and formally. Include a graphical representation of the states and transitions of your Turing machine. Be sure to specify all data.

Please provide a description of your Turing machine both in visual form (in the pdf) and as a `square.dat` file. It is strongly encouraged that you test your Turing machine extensively using your simulator, including on invalid inputs and corner cases.

Bookkeeping (5 points)

Indicate in a sentence or two how much time you spent on this homework, how difficult you found it subjectively, what you found to be the hardest part, how deeply you feel you understand the material it covers (0%–100%). Any non-empty answer to each of these questions will receive full credit.