Project 4: SAT Solvers

Spring 2022 - due May 13, 2020

Brian Lucero

David Cortez

Ricardo Barojas

Eduardo Romero

Your submission should contain SAT solver input and documentation as a file in`.pdf` format or in a tarball (`.tar.gz`, `.tgz`, `.tar.Z`, `.tar.bz2`, or `.tar.xz`) file containing the following items:

Chosen SAT solver: python Simple SAT Solver.

1. The input files in a format for use with your chosen SAT solver for each of the DFAs in Figures 6.3 and 6.4 of the textbook, along with inputs of length *t* = 9.

The following are the names of the input files for the use with our SAT Solver for each DFA:

1. DFA6-3.in
2. DFA6-4.in

These files can also be found at our GitHub Project Repository:
https://github.com/13rianlucero/439-Project4-SATSolvers

2. Output showing all satisfying assignments for each DFA.

The output for each DFA can be found in the output files.

1. DFA6-3.out
2. DFA6-4.out

These files can also be found at our GitHub Project Repository:
https://github.com/13rianlucero/439-Project4-SATSolvers

3. A description of how to run your chosen SAT solver and how to interpret the output to determine the accepted input strings.

This project used the python implementation of the SAT solver given in the project description. After downloading and opening the SAT solver file, download the provided DFA6-3 and DFA6-4 input and output files and add them to the projects directory under **/simple-sat-master/src/tests/simple**.
Next, open a terminal under **/simple-sat-master/src** and enter the commands:

```
python3 sat.py --all --brief --starting_with P  --input tests/simple/DFA6-3.in > tests/simple/DFA6-3.out
```

```
python3 sat.py --all --brief --starting_with P --input tests/simple/DFA6-4.in > tests/simple/DFA6-4.out
```

This will write the potential inputs for the DFA into their respective “.out” files.

When reading the files, each accepted input string is listed on a different line. The data listed in the line is listed as “P?” with the ? representing the position of any 1s present in the nine character input, with the rest of the slots being 0s. For example, “P1 P2 P4” means the full input string is “110100000” with only positions 1, 2, and 4 having a 1.