

# CS-557 Network Verification & Synthesis, Spring 2023

---

## Project: Z3-Sudoku

Files in archive

The following files are included in this archive:

```
|— README.md
|— README.pdf
|— requirements.txt
|— Z3_Solution.ipynb
|— Z3_Solution.py
|— Other_Solvers
    |— 16x16_sudoku_backtracking.py.time
    |— 16x16_sudoku_bitmasks.py.time
    |— 16x16_sudoku_naive.py.time
    |— 16x16_Z3_Solution.py.time
    |— 9x9_sudoku_backtracking.py.time
    |— 9x9_sudoku_bitmasks.py.time
    |— 9x9_sudoku_naive.py.time
    |— 9x9_Z3_Solution.py.time
    |— sudoku_backtracking.py
    |— sudoku_bitmasks.py
    |— sudoku_naive.py
    |— test_times.sh
```

File	Description
README.md	This README file as a Markdown file
README.pdf	This README file as a PDF file
requirements	Used by <code>pip</code> to install the Z3 solver and NumPy
Z3_Solution.ipynb	The Sudoku Z3 solution in a Jupyter Notebook
Z3_Solution.py	The Sudoku Z3 solution in a Python file
16x16_*.time	The execution time of all 4 Sudoku algorithms on a 16x16 puzzle, generated by <code>test_times.sh</code>
9x9_*.time	The execution time of all 4 Sudoku algorithms on a 9x9 puzzle, generated by <code>test_times.sh</code>
sudoku_backtracking.py	A Sudoku Back Tracking algorithm
sudoku_bitmasks.py	A Sudoku Back Tracking with Bit Masks algorithm
sudoku_naive.py	A Sudoku Naive algorithm

File	Description
test_times.sh	Runs all the Python files (only the ones with extension <code>.py</code> ) in batch and generate the execution time in milliseconds. Takes a required parameter, a string to prepend to each Python file and appends the string <code>.time</code> to each output file.

While a Jupyter Notebook file has been included (`Z3_Solution.ipynb`), it is not meant to be run. It includes additional markdown cells with stylized comments. The original source code in the Jupyter notebook can be found in the `Z3_Solution.py` file.

## Running the solution

The solution can be run using Virtual Environment (`virtualenv`) or using Anaconda.

### Using virtualenv

**Pre-requisites:** virtualenv and Python (3.10.6 or higher)

1. Create a new virtual environment: `virtualenv Z3_Sudoku` (you can use any name)
2. Activate the new environment: `source Z3_Sudoku/bin/activate` (or use the name you selected in step 1, if different than the default name)
3. Install the required modules: `pip install -r requirements.txt` (you must be in the root directory of this extracted archive)
4. Run the solution: `python3 Z3_Solution.py *`

### Using Anaconda

**Pre-requisites:** Anaconda (2023.3 or higher) and Python (3.10.6 or higher)

1. Create a new conda environment: `conda create --name Z3_Sudoku` (you can use any name)
2. Activate the new environment: `conda activate Z3_Sudoku` (or use the name you selected in step 1, if different than the default name)
3. Install the required modules: `pip install -r requirements.txt` (you must be in the root directory of this extracted archive)
4. Run the solution: `python3 Z3_Solution.py *`

\* If `python3` does not work on your system, the executable might have been symlink to `python` instead of `python3`.