ECE 448 / CS 440

Fall 2018

MP3

Assignment 3: CSP - Nonogram

Bo Wang
Yichen Zhou
Weili Liu

Part 1: Smaller Inputs

Algorithm:

We mainly used the backtrack DFS to find the solution. The variable we used are the rows and cols. The domain for each variable is all possible assignment for that individual variable given the constraint. We used arc constraint propagation before we enter the backtrack DFS to eliminate the unnecessary domains. We created a class called the nonogram_solver which stores the variable domains, and has functions to set and get the specific variables based on their types and index. Inside the backtrack_DFS function, every time we take out the most constraining variable. The nonogram_solver also has a priority queue for us to take out the most constraining variable. We loop through its domain and do forward checking for all the variable of the different type, ie.(if the current variable is row, we check all the columns, vice versa). We use forward checking to reduce the legal values for the variables that are yet to be assigned so the backtrack algorithm has less possibilities to go through in the future. This allows our algorithm to run a lot faster.

However, even with forward checking, using a backtracking DFS alone is still impossible to solve big puzzles. In order to make our implementation solve big puzzles in a reasonable amount of time. We did some preprocessing before entering backtracking algorithm. This preprocessing involves using the current domains of each variables to locate each inevitable colored or empty pixels (If every value in the domain of a variable has the same value for a specific pixel, then we know for sure the value for that pixel in the final solution). With this, we are able to fill in partial solutions for the puzzle. Then, using this partial solution, we then eliminate all values in the domain that are inconsistent with it and again use the new domain value to fill in partial solutions. This loop stops at 15 iterations or a complete solutions have been found. Finally, if a complete solution is not found when leaving the loop, we will use the backtracking DFS to find the rest of the solution. Adding this preprocessing increases the runtime for bigger inputs by a huge amount.

Output:

10_5:

```
[[0 1 1 0 0]
 [0 1 1 0 1]
 [0 0 1 0 1]
 [0 1 1 1 0]
 [1 0 1 0 0]
 [1 0 1 0 0]
 [0 0 1 1 0]
 [0 1 0 1 0]
 [0 1 0 1 1]
 [1 1 0 0 0]]
```

10_10_1:

```
[[0 0 1 0 1 0 1 1 1 1]
 [0 0 1 0 1 0 1 1 1 0]
 [0 1 1 0 0 0 1 1 1 1]
 [1 1 1 1 0 0 0 1 0 1]
 [1 1 1 0 0 0 0 1 1 1]
 [1 0 0 0 1 1 0 1 1 1]
 [1 0 0 0 0 0 0 1 1 1]
 [0 0 0 0 0 0 0 0 1 1]
 [1 1 1 0 0 0 0 0 0 0]
 [1 1 1 1 1 1 0 0 0 0]]
```

10_10_2:

```
[[0 0 1 1 1 1 1 1 0 0]
 [0 1 1 0 0 0 0 1 1 0]
 [0 1 0 0 0 0 0 0 1 0]
 [1 1 1 0 1 1 0 1 1 1]
 [1 0 0 1 1 1 1 0 0 1]
 [1 0 0 1 1 1 1 0 0 1]
 [1 1 1 0 1 1 0 1 1 1]
 [1 1 0 0 0 0 0 0 1 1]
 [0 1 1 0 0 0 0 1 1 0]
 [0 0 1 1 1 1 1 1 0 0]]
```

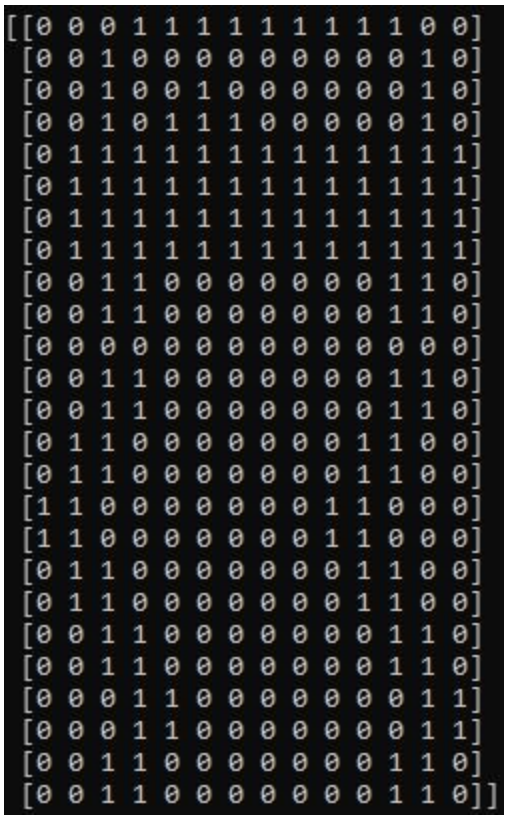
15_15_1:

```
[[1 0 0 1 1 1 1 1 0 0 0 0 0 1 1]
 [1 1 1 1 1 1 1 1 0 0 0 0 0 0 1]
 [0 1 1 1 1 1 1 1 0 0 0 0 0 0 1]
 [0 1 1 1 0 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 1 0 0 0 0 0 1 1 1 1 1]
 [0 0 1 1 0 0 0 0 0 1 1 1 1 1 0]
 [0 0 0 1 0 0 0 1 1 1 1 1 1 0 0]
 [0 0 0 0 0 0 0 1 1 1 1 1 1 0 0]
 [0 0 0 0 0 0 1 1 1 1 1 1 0 0 1]
 [0 0 0 0 0 0 0 1 1 1 1 1 0 0 0]
 [1 1 0 0 0 1 0 1 1 1 1 1 0 1 1]
 [1 1 0 0 0 1 0 0 1 1 1 0 1 1 1]
 [1 1 1 0 0 1 0 0 1 1 1 0 1 1 1]
 [1 1 1 0 0 0 0 0 0 0 0 0 1 1 1]
 [1 1 1 1 1 0 0 0 0 0 0 0 1 1 1]]
```

15_15_2:


[[1 1 0 1 0 1 0 1 0 1 1 1 1 1 1]
[1 1 0 0 0 0 0 0 0 1 1 1 1 1 1]
[1 1 0 1 0 1 0 1 0 1 1 1 1 1 1]
[1 0 0 1 1 1 0 1 1 1 1 1 1 1 1]
[0 0 0 1 1 1 1 1 1 1 1 1 0 0 1]
[0 0 0 1 1 1 1 1 1 1 1 0 0 0 0]
[0 1 1 1 1 1 1 1 1 1 1 0 0 0 0]
[0 1 1 1 1 0 0 1 1 0 1 0 1 1 1]
[0 0 0 0 0 0 0 0 1 0 1 1 1 1 1]
[0 0 0 0 0 0 1 0 0 0 0 0 0 1 1]
[0 0 0 0 0 0 1 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 1 0 0 0 0 0 0 0 0]
[0 0 0 0 0 1 1 0 0 1 1 1 0 0 0]
[0 0 0 0 1 1 1 1 1 1 1 1 0 1 0]
[0 0 0 0 1 1 1 1 1 1 1 1 0 0 0]]

14_25:


[[0 0 0 1 1 1 1 1 1 1 1 1 0 0]
[0 0 1 0 0 0 0 0 0 0 0 0 1 0]
[0 0 1 0 0 1 0 0 0 0 0 0 1 0]
[0 0 1 0 1 1 1 0 0 0 0 0 1 0]
[0 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 1 1 0 0 0 0 0 0 0 1 1 0 0]
[0 1 1 0 0 0 0 0 0 0 1 1 0 0]
[1 1 0 0 0 0 0 0 0 1 1 0 0 0]
[1 1 0 0 0 0 0 0 0 1 1 0 0 0]
[0 1 1 0 0 0 0 0 0 0 1 1 0 0]
[0 1 1 0 0 0 0 0 0 0 1 1 0 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 0 1 1 0 0 0 0 0 0 0 1 1]
[0 0 0 1 1 0 0 0 0 0 0 0 1 1]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]
[0 0 1 1 0 0 0 0 0 0 0 1 1 0]]

20_20:


[[0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0]
[1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0]
[1 0 0 0 0 1 1 1 1 1 1 1 0 0 0 1 0 0 0 1]
[1 0 0 0 0 1 1 1 1 1 1 1 1 0 0 1 1 0 1 1]
[1 0 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1]
[1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
[0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1]
[0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 1 1 1 0]
[0 0 1 1 1 1 1 0 0 0 1 1 1 1 0 0 0 0 0 0]
[0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0]
[0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0]
[0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
[0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
[0 0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0]
[0 0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0]]

35_25:


[[0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 1 1 0 1 1 0 0 0 0 0]
[0 0 0 0 0 0 0 1 1 1 0 1 1 0 1 1 1 0 1 0 1 1 1 0 0]
[0 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 0 1 1 1 0 0 1 1 0]
[0 0 0 0 1 0 0 0 1 0 1 0 0 0 1 1 1 1 1 1 0 1 0 1 0]
[0 0 0 1 1 0 0 1 1 0 0 1 1 0 0 0 1 1 0 0 1 1 0 1 0]
[0 0 0 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 0 1 1 0 0 1 0]
[0 0 1 1 0 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 1 1 0]
[0 0 1 0 0 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 0 0]
[0 0 0 1 1 1 0 0 0 0 0 0 0 1 1 1 0 0 1 1 1 1 1 0 0]
[0 0 0 1 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 1 1 0 0]
[0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 1 0 0 0 0 1 0 0 0]
[0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 0]
[0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0]
[0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0]
[0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 1 1 1 0 0 0 0 0 0 0]
[0 0 0 0 0 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0]
[0 0 0 0 0 0 1 0 0 0 1 1 0 0 1 1 1 1 1 1 1 0 1 0 0 1 0]
[0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1]
[0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1]
[0 1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 0 1 1 1 1 1 1 1 0]
[0 1 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 1 1 0]
[1 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 0 0 0 1 1 0 0]
[1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0]
[1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 1 1 1 1 0 0 0 0]
[1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 1 1 1 0 1 1 1 1 0 0 1 1 0 1 0 0]
[0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0]
[0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0]]

23_27:

```
[[0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0]
 [0 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0 0]
 [0 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 1 0 0 0 0 0]
 [0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 0 0]
 [1 1 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 0 0 0]
 [1 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 1 1 0 0 0 0]
 [1 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 0 0]
 [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0]
 [1 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 1 0 1 1 0 1 0 0 0 0]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 1 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 0 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 1 1]
 [1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 0 0]]
```

Runtime:

10_5:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/10_5.npy
[[0 1 1 0 0]
 [0 1 1 0 1]
 [0 0 1 0 1]
 [0 1 1 1 0]
 [1 0 1 0 0]
 [1 0 1 0 0]
 [0 0 1 1 0]
 [0 1 0 1 0]
 [0 1 0 1 1]
 [1 1 0 0 0]]
runtime is:  0.01081395149230957 sec
Success!
```

10_10_1:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/10_10_1.npy
[[0 0 1 0 1 0 1 1 1 1]
 [0 0 1 0 1 0 1 1 1 0]
 [0 1 1 0 0 0 1 1 1 1]
 [1 1 1 1 0 0 0 1 0 1]
 [1 1 1 0 0 0 0 1 1 1]
 [1 0 0 0 1 1 0 1 1 1]
 [1 0 0 0 0 0 0 1 1 1]
 [0 0 0 0 0 0 0 0 1 1]
 [1 1 1 0 0 0 0 0 0 0]
 [1 1 1 1 1 1 0 0 0 0]]
runtime is:  0.018934965133666992 sec
Success!
```

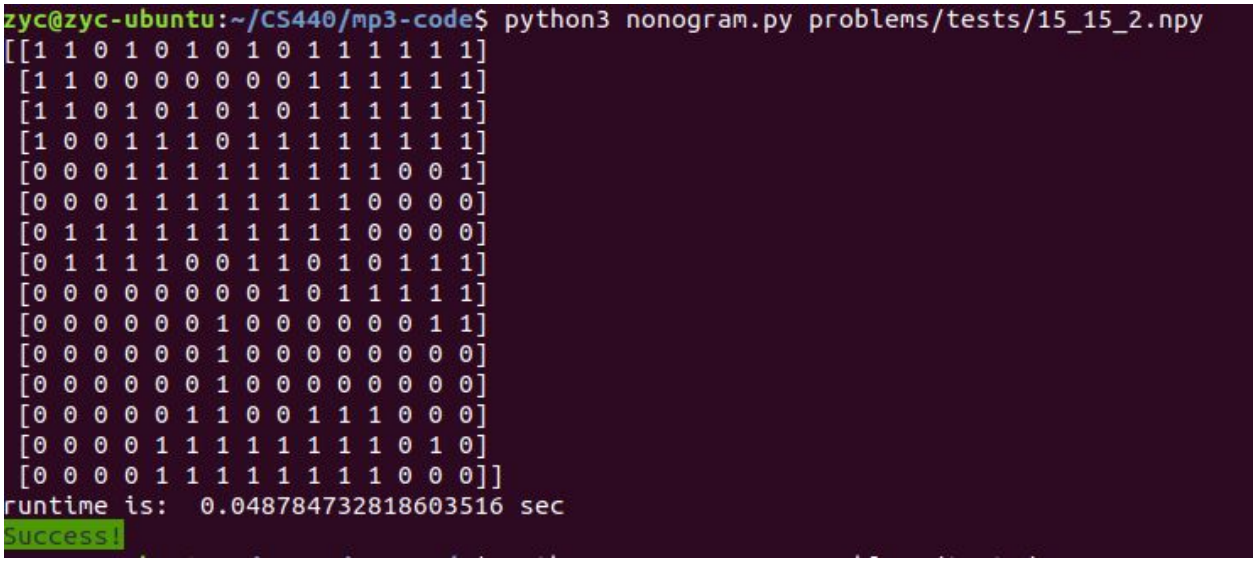
10_10_2:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/10_10_2.npy
[[0 0 1 1 1 1 1 1 0 0]
 [0 1 1 0 0 0 0 1 1 0]
 [0 1 0 0 0 0 0 0 1 0]
 [1 1 1 0 1 1 0 1 1 1]
 [1 0 0 1 1 1 1 0 0 1]
 [1 0 0 1 1 1 1 0 0 1]
 [1 1 1 0 1 1 0 1 1 1]
 [1 1 0 0 0 0 0 0 1 1]
 [0 1 1 0 0 0 0 1 1 0]
 [0 0 1 1 1 1 1 1 0 0]]
runtime is:  0.01519322395324707 sec
Success!
```

15_15_1:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/15_15_1.npy
[[1 0 0 1 1 1 1 1 0 0 0 0 0 1 1]
 [1 1 1 1 1 1 1 1 0 0 0 0 0 0 1]
 [0 1 1 1 1 1 1 1 0 0 0 0 0 0 1]
 [0 1 1 1 0 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 1 0 0 0 0 0 1 1 1 1 1]
 [0 0 1 1 0 0 0 0 0 1 1 1 1 1 0]
 [0 0 0 1 0 0 0 1 1 1 1 1 1 0 0]
 [0 0 0 0 0 0 0 1 1 1 1 1 1 0 0]
 [0 0 0 0 0 0 1 1 1 1 1 1 0 0 1]
 [0 0 0 0 0 0 0 1 1 1 1 1 0 0 0]
 [1 1 0 0 0 1 0 1 1 1 1 1 0 1 1]
 [1 1 0 0 0 1 0 0 1 1 1 0 1 1 1]
 [1 1 1 0 0 1 0 0 1 1 1 0 1 1 1]
 [1 1 1 0 0 0 0 0 0 0 0 0 1 1 1]
 [1 1 1 1 1 0 0 0 0 0 0 0 1 1 1]]
runtime is:  0.10271787643432617 sec
Success!
```

15_15_2:



Part 2: Bigger Inputs

Runtime:

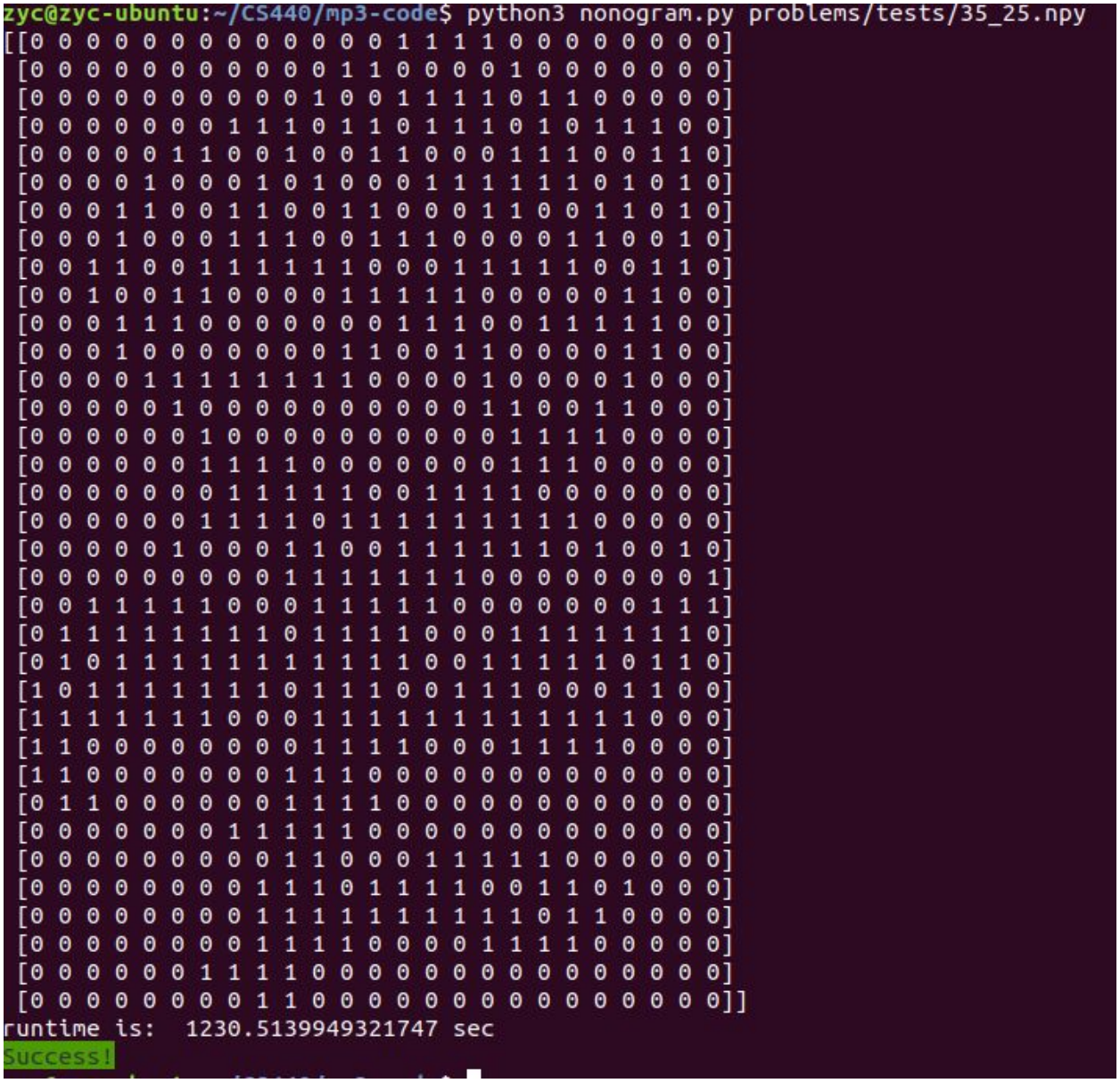
14_25:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/14_25.npy
[[0 0 0 1 1 1 1 1 1 1 1 1 0 0]
 [0 0 1 0 0 0 0 0 0 0 0 0 1 0]
 [0 0 1 0 0 1 0 0 0 0 0 0 1 0]
 [0 0 1 0 1 1 1 0 0 0 0 0 1 0]
 [0 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 1 1 0 0 0 0 0 0 0 1 1 0 0]
 [0 1 1 0 0 0 0 0 0 0 1 1 0 0]
 [1 1 0 0 0 0 0 0 0 0 1 1 0 0 0]
 [1 1 0 0 0 0 0 0 0 0 1 1 0 0 0]
 [0 1 1 0 0 0 0 0 0 0 1 1 0 0]
 [0 1 1 0 0 0 0 0 0 0 1 1 0 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 0 1 1 0 0 0 0 0 0 0 1 1]
 [0 0 0 1 1 0 0 0 0 0 0 0 1 1]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]
 [0 0 1 1 0 0 0 0 0 0 0 1 1 0]]
runtime is:  0.9398200511932373 sec
Success!
```

20_20:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/20_20.npy
[[0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0]
 [1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0]
 [1 0 0 0 0 1 1 1 1 1 1 1 0 0 0 1 0 0 0 1]
 [1 0 0 0 0 1 1 1 1 1 1 1 1 0 0 1 1 0 1 1]
 [1 0 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1]
 [1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
 [0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1]
 [0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 1 1 1 0]
 [0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0]
 [0 0 0 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0]
 [0 0 0 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
 [0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0]]
runtime is:  0.45781803131103516 sec
Success!
```

35_25:



Extra Credit:

23_27:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/23_27.npy
[[0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0]
 [0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0]
 [0 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0 0]
 [0 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 1 0 0 0 0 0]
 [0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 0 0]
 [1 1 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 1 0 1 1 1 0 0 0 0]
 [1 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 1 1 0 0 0 0]
 [1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 0 0 0]
 [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0 0]
 [1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0 0]
 [1 0 0 0 1 1 1 1 1 1 1 1 1 1 0 0 0 1 0 1 1 0 1 0 0 0 0]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 1 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 0 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 1]
 [1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 1 1]
 [1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 1 1 1 0 0]]
runtime is:  50.68739295005798 sec
Success!
```

34_34:

```
zyc@zyc-ubuntu:~/CS440/mp3-code$ python3 nonogram.py problems/tests/34_34.npy
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
runtime is:  2644.71874332428 sec
Success!
```

Why will runtimes of puzzles of the same size differ?

The runtimes of puzzles of the same size differ because the one with less constraints will have a larger domain. A larger domain means that it takes longer to be generated and to be preprocessed. Also, a larger domain also means the DFS will have to go through more values before finding the final solution. Therefore, the puzzle with less constraints will have longer runtime.