

In order to run the program, run the function `solve(inputfile, outputfile)`. The parameters are the input file with the uncompleted matrix and the output file is where the solution will write to.

Warning: the program takes a pretty long time to run

As a CSP, the variables are $X_1 \dots X_{81}$ to represent each cell. The domains of each variable are $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ to represent all possible values of the cell. The constraints are that each cell cannot have the same value as another cell in the same row, column, or square.

Input1:

```
0 0 4 6 0 0 9 0 0
0 1 0 0 0 8 0 0 7
0 0 0 0 2 0 0 0 3
8 9 0 0 0 0 0 0 0
3 0 0 0 0 0 0 2 0
0 4 0 0 0 6 0 0 5
7 0 0 0 0 0 0 4 0
0 5 9 0 0 0 2 0 0
0 0 0 0 0 0 3 0 0
```

Input2:

```
0 0 0 0 0 0 0 0 0
0 0 0 8 0 1 0 0 0
0 0 0 0 2 0 0 0 0
0 0 3 0 0 8 6 0 4
0 0 6 0 9 0 0 0 3
0 0 0 5 0 0 0 0 0
0 4 0 2 0 0 0 0 0
9 6 8 3 5 0 0 0 0
0 7 0 0 0 0 3 0 6
```

Input3:

```
0 0 0 1 0 3 0 0 0
0 0 8 5 6 4 0 0 0
0 0 0 7 0 0 0 2 0
7 9 0 0 0 0 5 8 0
0 0 4 0 0 0 1 0 0
0 0 0 0 0 5 0 0 0
0 0 6 0 0 7 0 0 0
0 7 0 0 0 0 2 0 0
8 0 0 3 0 0 0 5 0
```

Output1:

```
5 3 4 6 7 1 9 8 2
9 1 2 3 5 8 4 6 7
6 8 7 4 2 9 5 1 3
8 9 6 5 1 2 7 3 4
3 7 5 9 8 4 6 2 1
2 4 1 7 3 6 8 9 5
7 6 3 2 9 5 1 4 8
1 5 9 8 4 3 2 7 6
4 2 8 1 6 7 3 5 9
```

Output2:

```
8 3 2 9 7 4 1 6 5
6 5 4 8 3 1 7 2 9
7 1 9 6 2 5 4 3 8
5 2 3 7 1 8 6 9 4
1 8 6 4 9 2 5 7 3
4 9 7 5 6 3 8 1 2
3 4 1 2 8 6 9 5 7
9 6 8 3 5 7 2 4 1
2 7 5 1 4 9 3 8 6
```

Output3:

```
9 6 7 1 2 3 8 4 5
2 1 8 5 6 4 3 7 9
5 4 3 7 8 9 6 2 1
7 9 2 6 4 1 5 8 3
6 5 4 8 3 2 1 9 7
3 8 1 9 7 5 4 6 2
4 3 6 2 5 7 9 1 8
1 7 5 4 9 8 2 3 6
8 2 9 3 1 6 7 5 4
```