SUDOKU

Dylan Lawler, Eric Leung, Jordan Kirkbride, Alex Jones

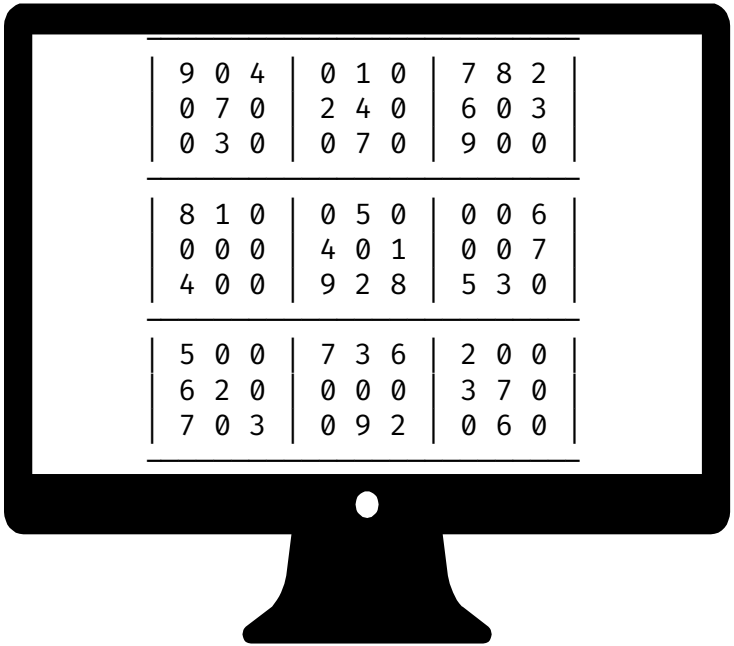
Board Struct

```
{0=2, 1=1, 2=9, 3=0, 4=7, 5=0, 6=3, 7=0, 8=0, }
{0=0, 1=4, 2=5, 3=9, 4=3, 5=8, 6=0, 7=1, 8=6, }
{0=8, 1=6, 2=0, 3=0, 4=4, 5=0, 6=7, 7=0, 8=0, }
{0=5, 1=0, 2=0, 3=0, 4=8, 5=0, 6=0, 7=6, 8=7, }
{0=4, 1=0, 2=8, 3=6, 4=0, 5=0, 6=1, 7=0, 8=0, }
{0=0, 1=7, 2=0, 3=0, 4=1, 5=5, 6=8, 7=2, 8=0, }
{0=1, 1=0, 2=4, 3=0, 4=2, 5=0, 6=0, 7=8, 8=0, }
{0=6, 1=0, 2=2, 3=8, 4=0, 5=0, 6=4, 7=0, 8=0, }
{0=3, 1=0, 2=0, 3=1, 4=6, 5=0, 6=0, 7=9, 8=0, }
```

Array of Counters

- Similar to Index struct
- Removed hash aspect to prevent randomness
- Key is column number
- Value is cell number

Board Struct



Module Functions

- insert_number()
- get_number/row/size()
- save_solution()
- copy_board()
- print_board()

Loading from Stdin

Reading from stdin

- Read character by character
- Required space after number to count as valid
- Keep track of previous & current characters

```
$sudoku solve
Enter sudoku board:
```

Error catching invalid scenarios (total value count, alphabetical, row/column #s, >1 digit # when not allowed)

Making and Solving Algorithms

- Return true if the board is full
- For every possible number to input (shuffled when making):
  - if the number is allowed
    - Insert it
    - Recursively go to the next cell and do the same
  - No numbers worked put 0 back
  - Return false

Solved unique Sudoku board:

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

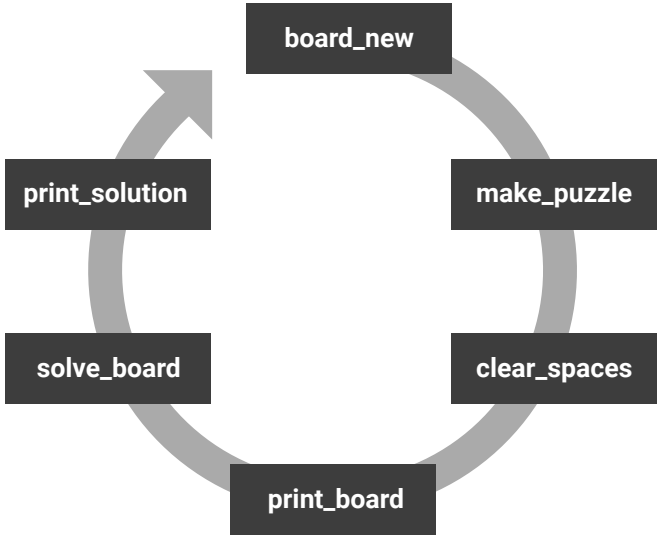
Unique Solutions

```
int sols = solve_puzzle(board, 0, 0, 0);
```

- Solve function keeps track of how many solutions have been found
- clear_spaces ensures a unique solution by calling solve at each step
- Solution saved in board struct in solve function

Testing - fuzztest.c

n iterations:




```
$ ./fuzz 1
UNSOLVED
 ---------------------
| 3 0 8 | 4 0 5 | 6 2 0 |
| 7 0 5 | 6 3 9 | 4 1 0 |
| 6 0 4 | 8 0 0 | 3 0 0 |
 ---------------------
| 4 0 0 | 0 7 0 | 0 0 2 |
| 0 8 0 | 3 5 0 | 7 0 0 |
| 0 0 7 | 0 0 0 | 0 6 0 |
 ---------------------
| 5 0 1 | 0 8 0 | 0 0 6 |
| 9 7 0 | 0 6 4 | 0 8 3 |
| 0 0 3 | 7 9 0 | 2 0 5 |
 ---------------------
```

```
SOLVED
 ---------------------
| 3 9 8 | 4 1 5 | 6 2 7 |
| 7 2 5 | 6 3 9 | 4 1 8 |
| 6 1 4 | 8 2 7 | 3 5 9 |
 ---------------------
| 4 5 9 | 1 7 6 | 8 3 2 |
| 1 8 6 | 3 5 2 | 7 9 4 |
| 2 3 7 | 9 4 8 | 5 6 1 |
 ---------------------
| 5 4 1 | 2 8 3 | 9 7 6 |
| 9 7 2 | 5 6 4 | 1 8 3 |
| 8 6 3 | 7 9 1 | 2 4 5 |
 ---------------------
```

Testing - testing.sh

Argument Input

- No arguments

Create

- Make a random board

Solve

- num_of_puzzle_input > 1
- Invalid board input
  - size
  - [a-zA-Z]
  - entry number > 9
  - empty

Extra credit: 4x4, 16x16 Sudoku

```
Enter Sudoku board:
-------------
| 2 3 | 1 4 |
| 0 0 | 0 0 |
-------------
| 1 0 | 2 0 |
| 0 2 | 0 1 |
-------------
Solved Sudoku board:
-------------
| 2 3 | 1 4 |
| 4 1 | 3 2 |
-------------
| 1 4 | 2 3 |
| 3 2 | 4 1 |
-------------
```

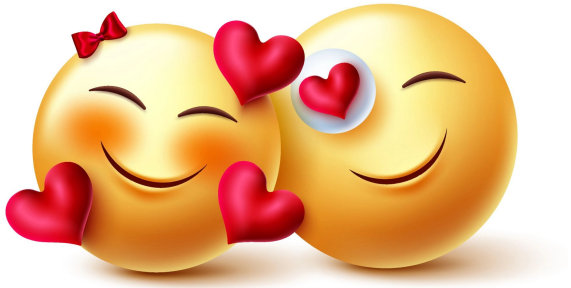
```
Enter Sudoku board:
-----------------------------------------------------------------------
| 0  0  0  12 | 2  0  0  16 | 0  10 0  6  | 0  9  7  13 |
| 6  0  0  0  | 11 0  3  8  | 0  9  7  0  | 14 0  0  12 |
| 10 0  0  7  | 0  15 0  0  | 0  2  0  4  | 0  0  0  0  |
| 11 15 4  13 | 0  0  0  0  | 0  14 0  0  | 0  1  2  0  |
-----------------------------------------------------------------------
| 0  5  0  0  | 13 11 1  0  | 0  0  8  0  | 0  0  12 2  |
| 0  0  0  6  | 0  0  0  2  | 0  11 0  0  | 8  3  0  15 |
| 0  0  0  0  | 4  8  16 0  | 0  0  0  12 | 0  6  5  0  |
| 0  8  16 0  | 0  0  0  0  | 10 0  2  3  | 7  0  0  0  |
-----------------------------------------------------------------------
| 0  0  10 4  | 0  0  15 5  | 6  12 11 2  | 0  14 0  8  |
| 15 6  0  0  | 14 2  10 12 | 1  0  13 8  | 4  0  9  0  |
| 14 9  0  11 | 0  0  13 0  | 0  0  0  0  | 0  0  1  5  |
| 1  0  12 0  | 8  7  11 4  | 0  3  0  0  | 6  15 10 16 |
-----------------------------------------------------------------------
| 0  0  6  3  | 0  0  0  0  | 14 0  10 9  | 12 16 15 7  |
| 0  16 15 14 | 3  4  0  9  | 0  0  0  7  | 0  0  0  10 |
| 7  12 13 9  | 15 0  8  14 | 0  1  16 11 | 5  2  0  0  |
| 2  10 0  0  | 6  16 0  0  | 12 0  4  15 | 0  13 0  0  |
-----------------------------------------------------------------------
Solved Sudoku board:
-----------------------------------------------------------------------
| 8  3  5  12 | 2  14 4  16 | 11 10 1  6  | 15 9  7  13 |
| 6  1  2  16 | 11 5  3  8  | 15 9  7  13 | 14 10 4  12 |
| 10 14 9  7  | 1  15 6  13 | 16 2  12 4  | 11 5  8  3  |
| 11 15 4  13 | 7  12 9  10 | 8  14 3  5  | 16 1  2  6  |
-----------------------------------------------------------------------
| 3  5  14 15 | 13 11 1  7  | 9  6  8  16 | 10 4  12 2  |
| 12 4  7  6  | 10 9  14 2  | 13 11 5  1  | 8  3  16 15 |
| 9  2  11 10 | 4  8  16 3  | 7  15 14 12 | 13 6  5  1  |
| 13 8  16 1  | 12 6  5  15 | 10 4  2  3  | 7  11 14 9  |
-----------------------------------------------------------------------
| 16 7  10 4  | 9  1  15 5  | 6  12 11 2  | 3  14 13 8  |
| 15 6  3  5  | 14 2  10 12 | 1  16 13 8  | 4  7  9  11 |
| 14 9  8  11 | 16 3  13 6  | 4  7  15 10 | 2  12 1  5  |
| 1  13 12 2  | 8  7  11 4  | 5  3  9  14 | 6  15 10 16 |
-----------------------------------------------------------------------
| 4  11 6  3  | 5  13 2  1  | 14 8  10 9  | 12 16 15 7  |
| 5  16 15 14 | 3  4  12 9  | 2  13 6  7  | 1  8  11 10 |
| 7  12 13 9  | 15 10 8  14 | 3  1  16 11 | 5  2  6  4  |
| 2  10 1  8  | 6  16 7  11 | 12 5  4  15 | 9  13 3  14 |
-----------------------------------------------------------------------
```

Lessons Learned

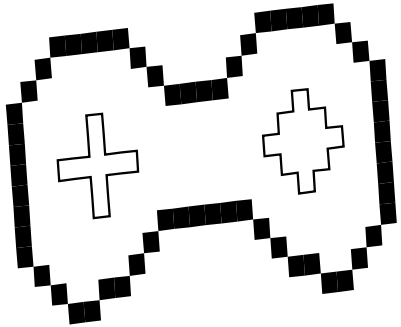
- GitHub management
- Debugging and integrating each other’s code
- Staying calm when running into errors

But most importantly…

The ***Friends*** we Made Along the Way <3



?



You are amazing :)

Thanks!