# Crossword Generator

GROUP MEMBERS:
Owais Ullah Faiz
Naveed Ahmad

# What is Cross Word Generator?

- A program that uses models the crossword generation as CSP and solves it via backtracking with a few heuristics.
- It takes as input two txt files: words list and structure of crossword.
- The generaator prints a crossword grid based on the input data.

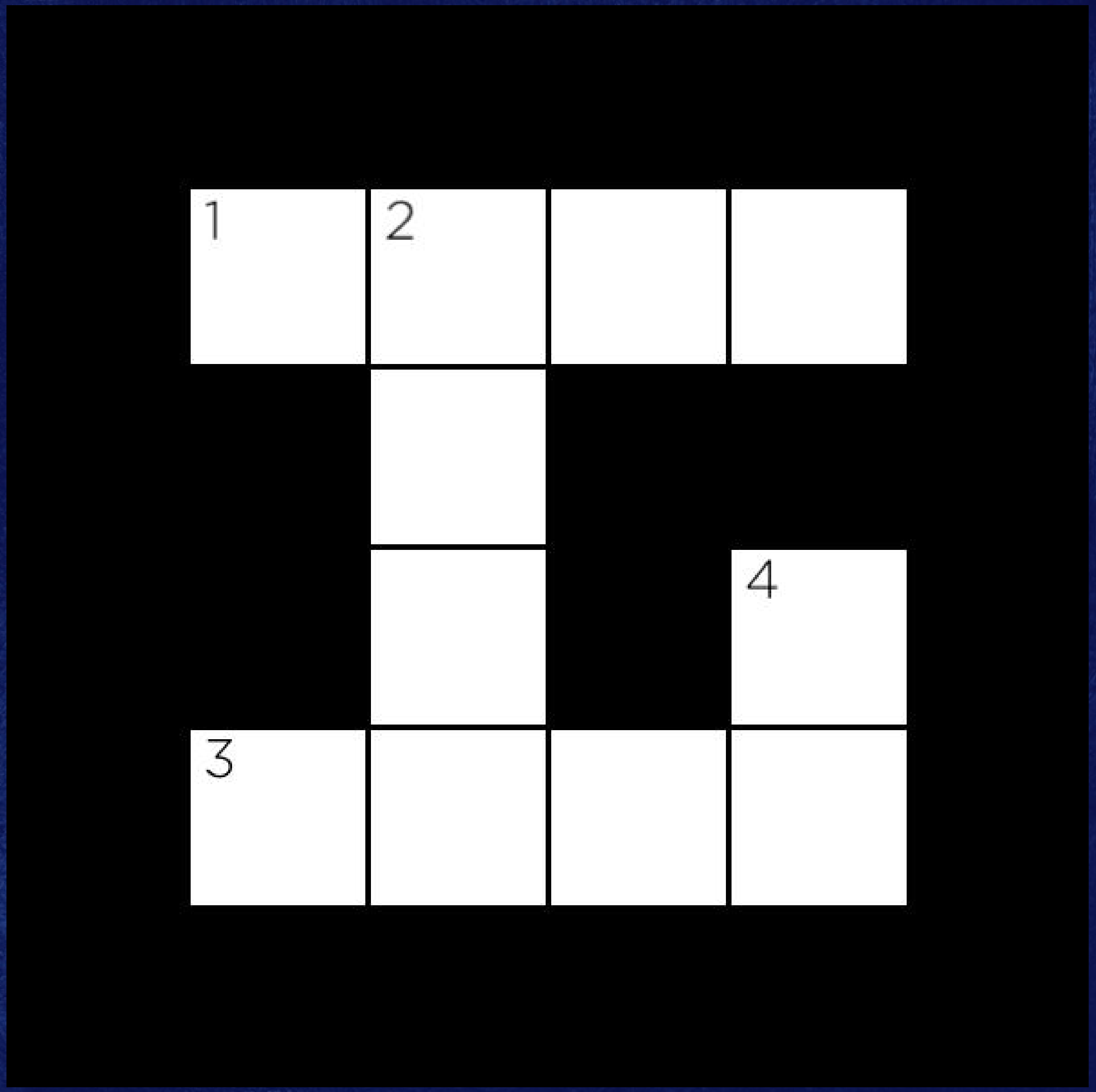
# Base project

- From CS50 AI edx course.
- It had two defined functions:
  Variable
  Crossword
- These gave an idea of how to read the files and define varibles for the CSP.
- We did the rest i.e defining domains, node consistency, arc consistency, backtracking, heuristics, output.

# The model

- Variable definition:
  1. i value: row number of starting cell.
  2. j value: column number of starting cell.
  3. Length of word.
  4. Direction of word.
- Domain:

  All the words in the given data.



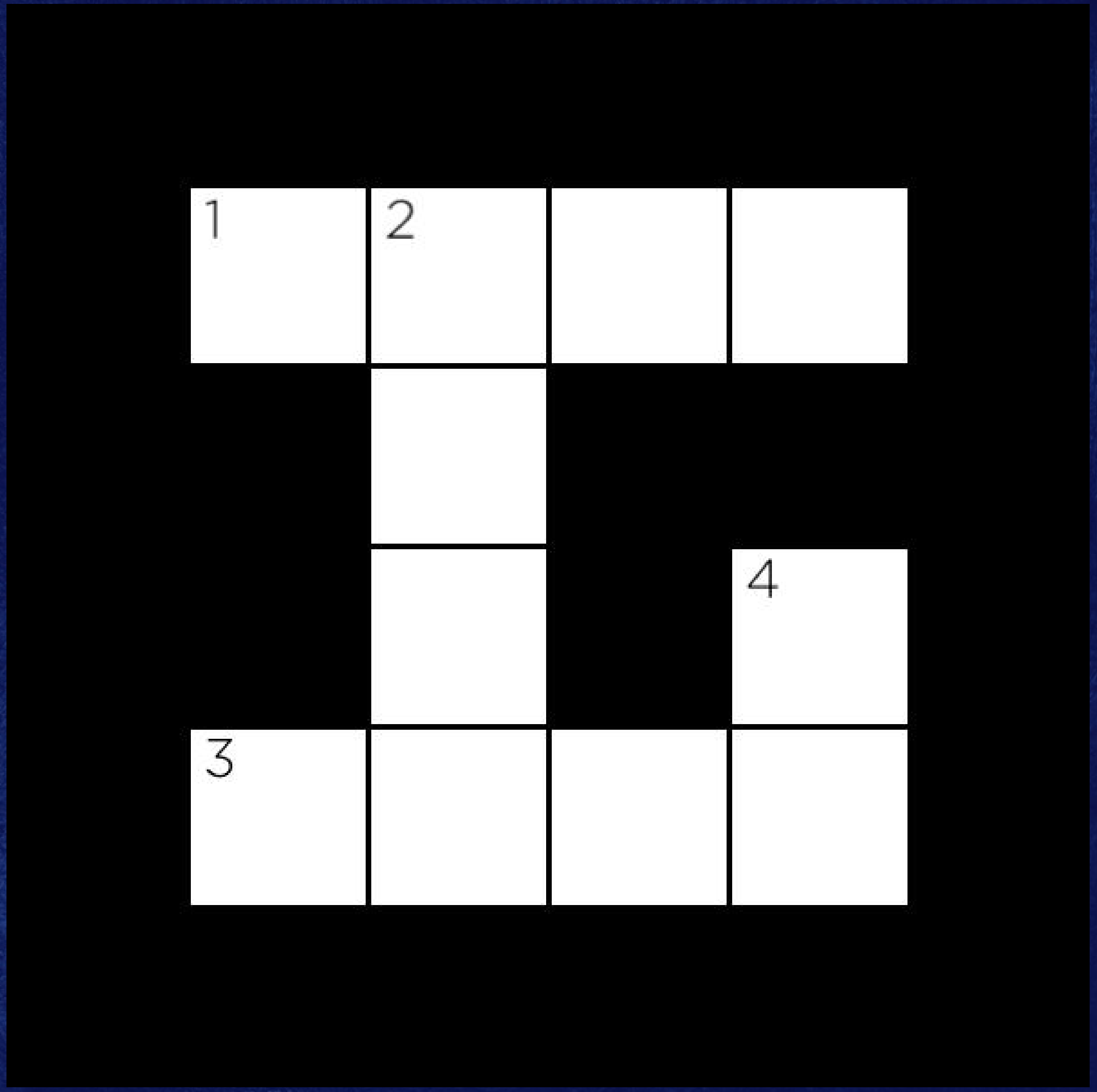
# The model

- Constraints:
    1. Unary:

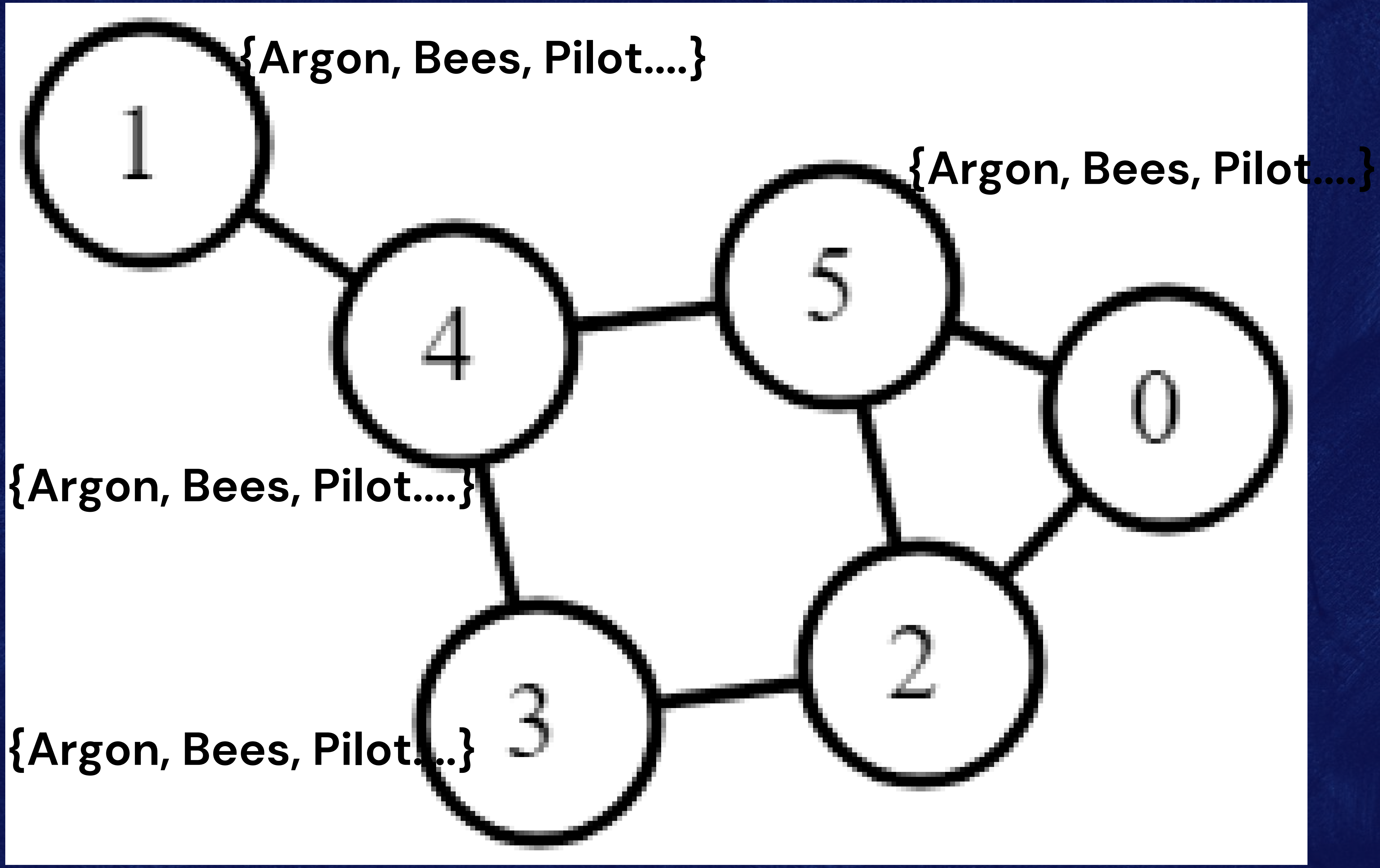
       length of word = variable length.
    2. Binary:

       Overlap between words.

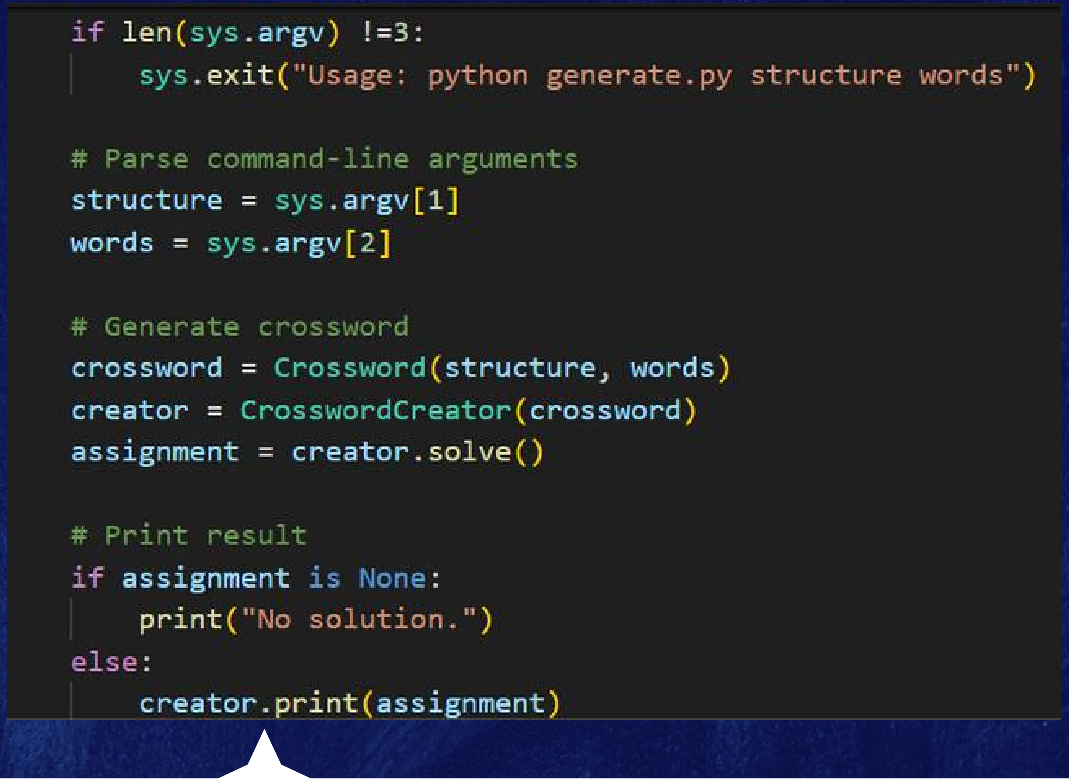
       No repitition.



# The model Flow Chart



# Flowchart



```
if len(sys.argv) !=3:
    sys.exit("Usage: python generate.py structure words")

# Parse command-line arguments
structure = sys.argv[1]
words = sys.argv[2]

# Generate crossword
crossword = Crossword(structure, words)
creator = CrosswordCreator(crossword)
assignment = creator.solve()

# Print result
if assignment is None:
    print("No solution.")
else:
    creator.print(assignment)
```

## Main:

01

Check and parse command line arguments.

02

Create Crossword

03

Solve

04

Print

# Functions

## Crossword

01

Structure Initialization: 2D list self.structure.

02

Vocabulary Loading: Set Self.words

03

Variable detection: Set self.varibles

04

Overlap Calculation: Dictionary self.overlaps

# Functions

## Crosswordcreator

01

Initialization:

- generate new crossword.

02

letter_grid:

- Converts the assignment of words to a 2D grid of letters

03

print:

- Prints the current state of the crossword puzzle to the terminal.

04

solve:

- Solves the crossword using constraint satisfaction techniques.

# Functions

## Crosswordcreator

5

enforce_node_consistency:

- unary constraints satisfaction

6

revise:

- Makes one variable arc-consistent with another.

7

ac3:

- Enforces arc consistency across all variables.

8

assignment_complete:

- Checks if the assignment is complete.

# Functions

## Crosswordcreator

### 9

consistent:

- Checks if the current assignment is consistent.

### 10

order_domain_values:

- Orders the domain values for a variable. (least constraining)

### 11

select_unassigned_variable:

- Selects an unassigned variable. (shortest domain)

### 12

backtrack:

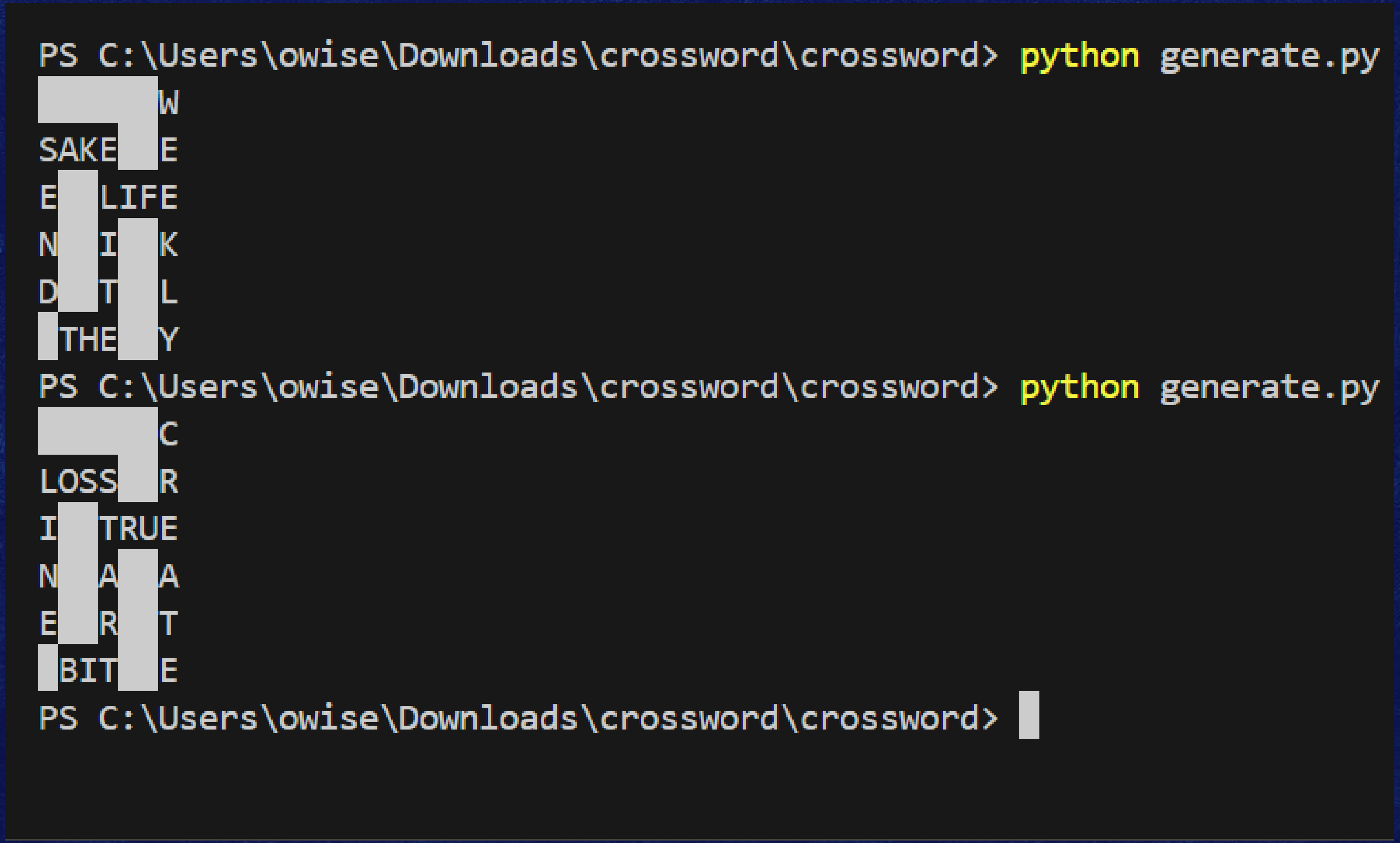
- Uses backtracking to find a complete and consistent assignment.

# Functions

## Backtracking

```
if assignment complete:
    return assignment
var = Select-Unassigned-Var(assignment, csp)
for value in Domain-Values(var, assignment, csp):
    if value consistent with assignment:
        add {var = value} to assignment
        result = Backtrack(assignment, csp)
        if result ≠ failure:
            return result
        remove {var = value} from assignment
return failure
```

# Output



```
PS C:\Users\owise\Downloads\crossword\crossword> python generate.py
█████W
SAKE█E
E█LIFE
N█I██K
D█T██L
█THE█Y
PS C:\Users\owise\Downloads\crossword\crossword> python generate.py
█████C
LOSS█R
I█TRUE
N█A██A
E█R██T
█BIT█E
PS C:\Users\owise\Downloads\crossword\crossword> 
```

# Conclusion

- Errors:

  Backtracking recursion function.

  Randomness

  Saving to image file

- Link:

  https://github.com/Owais-Faiz/Project-AI.git

# Thank you very much!