SOLVING THE PADLOCK PROBLEM

- Using Word Combinations to Help Farmer John
- Presented by: Naveen Kumar Kottidi

INTRODUCTION TO THE PROBLEM

- Problem Overview:
  - Farmer John has a special lock with 5 slots.
  - Each slot has 3 possible letters.

- Goal:
  - Maximize the number of valid words formed by choosing the best third row word.

UNDERSTANDING THE COMBINATION PROCESS

Combination Generation and Word Creation

Watch how rotating each slot creates new combinations
Click this link for the Video

Video Source: YouTube | Whole Brain Escape: Raleigh Escape Room- Open a Letter Lock.

THE LOCK SETUP

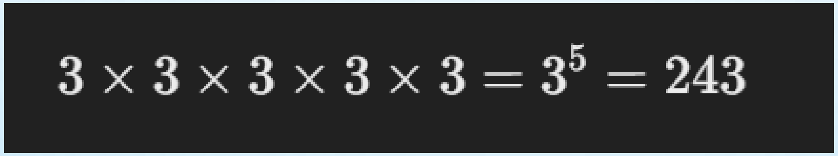
- Fixed Rows:
  - Row 1: BELLA
  - Row 2: LOVES
- Flexible Row:
  - Row 3: Farmer John can choose any word.
- Challenge: Find the best third word to create as many valid word combinations as possible.

UNDERSTANDING THE COMBINATION PROCESS

Step 1: Fix Rows 1 and 2.

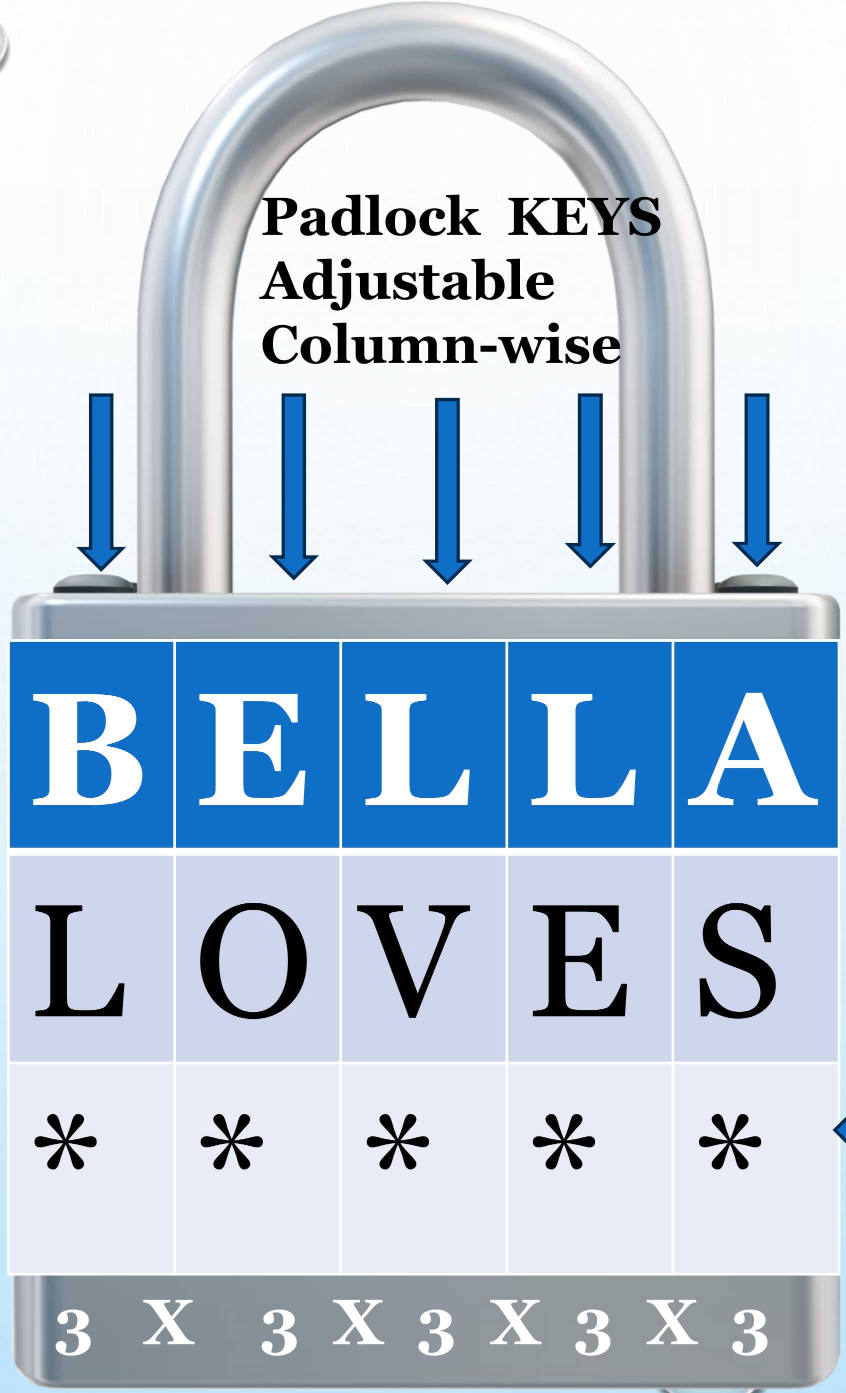
Step 2: Choose a candidate word for Row 3.

Step 3: Rotate each letter slot to make different word combinations.



$$3 \times 3 \times 3 \times 3 \times 3 = 3^5 = 243$$

3 options per slot → 243 total combinations.



Padlock KEYS
Adjustable
Column-wise

Row1 → Fixed	B	E	L	L	A
Row2 → Fixed	L	O	V	E	S
Row3 → Flexible	*	*	*	*	*

Possible Combinations =243→ 3 X 3 X 3 X 3 X 3

Find this word from the list that maximizes the valid word combinations for the padlock

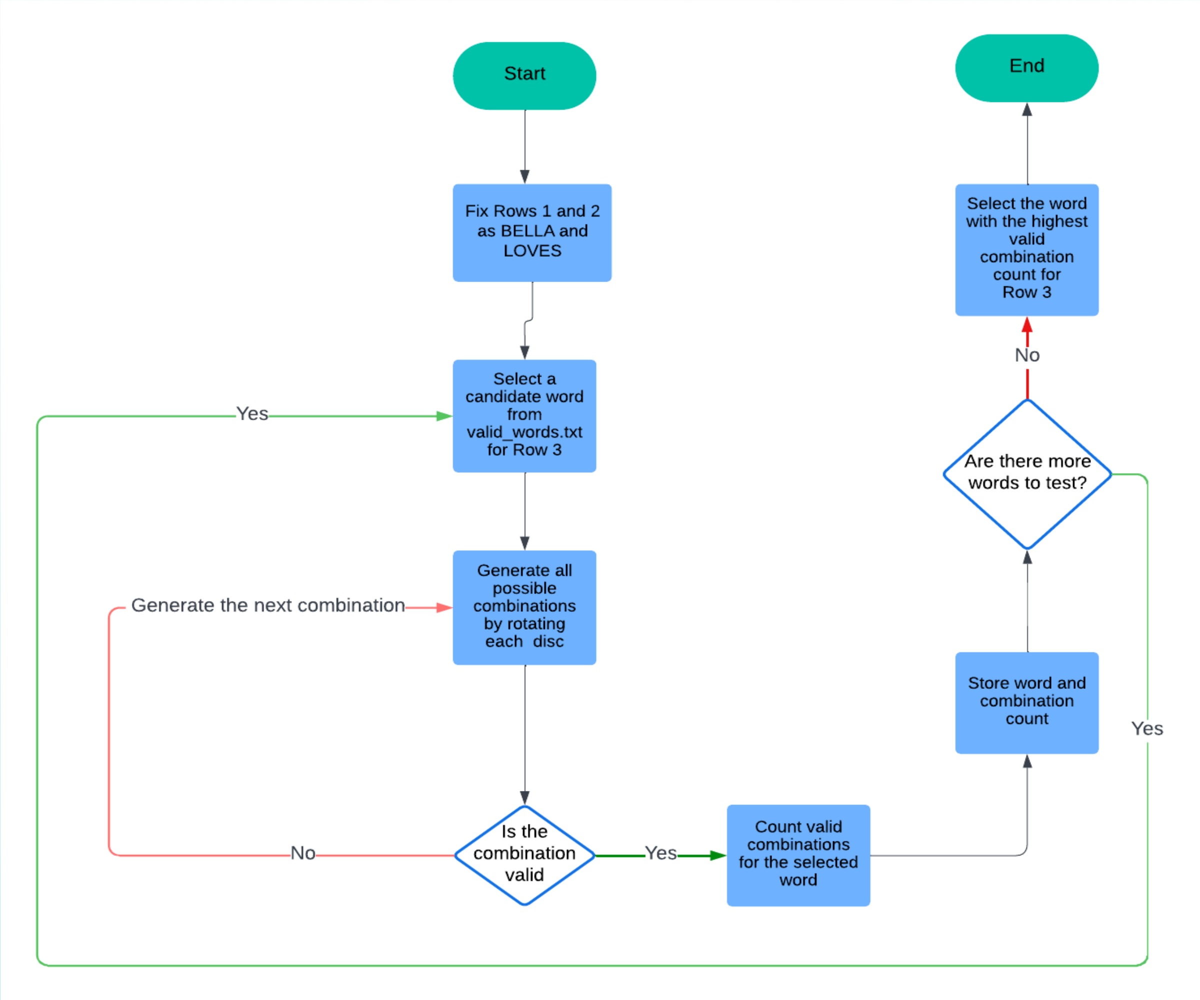
SOLUTION OUTLINE

Goal: Count all valid word combinations for each candidate word.

Steps:

1. Select a candidate word for Row 3.
2. Generate all possible combinations.
3. Check each combination against the list of valid words.(given)
4. Count the valid combinations.
5. Repeat for each candidate and pick the best.

SOLUTION OUTLINE: PADLOCK PROBLEM FLOWCHART


Start
Fix Rows 1 and 2 as BELLA and LOVES
Select a candidate word from valid_words.txt for Row 3
Generate all possible combinations by rotating each disc
Is the combination valid
Yes
Count valid combinations for the selected word
No
Generate the next combination
Store word and combination count
Are there more words to test?
Yes
Yes
No
Select the word with the highest valid combination count for Row 3
End

Using Lucid Charts

Python Code to generate combinations*(without using `itertools`)*

```
def generate_combinations(flexible_row):
    combinations = []
    for i in range(3):  # Rotate first letter position
        for j in range(3):  # Rotate second letter position
            for k in range(3):  # Rotate third letter position
                for l in range(3):  # Rotate fourth letter position
                    for m in range(3):  # Rotate fifth letter
                      for m in range(3):  # Rotate fifth letter position
                # Create a new combination by modifying each position
                          new_word = (
                              flexible_row[0][i] +
                              flexible_row[1][j] +
                              flexible_row[2][k] +
                              flexible_row[3][l] +
                              flexible_row[4][m]
                          )
                          combinations.append(new_word)
      return combinations
```

Python Code to check validity:

```
def count_valid_combinations(flexible_row, valid_words):
    valid_count = 0
    combinations = generate_combinations(flexible_row)
    for word in combinations:
        if word in valid_words:
            valid_count += 1
    return valid_count
```

Python Code to find the optimal word:

```
best_word = ""
max_valid_count = 0

for word in valid_words:
    valid_count = count_valid_combinations(word, valid_words)
    if valid_count > max_valid_count:
        best_word = word
        max_valid_count = valid_count
```

EXAMPLE: TRYING 'SHEDS' AS THE THIRD ROW

- **Row Setup:**
  - Row 1: BELLA
  - Row 2: LOVES
  - Row 3: SHEDS
- **Generated Words:**
  - Example valid words: BELLS, SOLES, SHEDS.
- **Result:** Count how many valid words we get.

KEY CONCEPTS

- **Combination Generation:**
  - Rotating letters to create new words.

- **Valid Word Check:**
  - Use a word list to verify if generated words are real.

- **Optimization:**
  - Only count combinations that appear in the valid words list.

FINAL SOLUTION

- **Result**: Choose the candidate that produces the most valid words.
- **Outcome**: Farmer John can now maximize his word options!

INTERACTIVE CHALLENGE: EXPLORING ALTERNATIVES

- *“What would happen if Farmer John chose a different third word?”*

- *“Can you think of a third word that might generate even more valid combinations?”*

OPTIONAL EXTENSIONS: GOING BEYOND THE PROBLEM

Determine the Best N Words

- *Challenge*: Given an integer N, find the top N words that maximize valid word combinations.
- *Goal*: Identify multiple optimal words, not just the single best, for maximizing combinations.

Add a Fourth Row

- *Challenge*: Add a fourth row of letters to the lock.
- *Goal*: Choose two extra words for Rows 3 and 4 that together allow for the maximum number of valid combinations.

REAL-WORLD APPLICATIONS OF COMBINATION GENERATION

Code Encryption:

- Uses secure combinations of characters to protect information, making it difficult for others to guess without the exact combination.

Security Systems:

- Locks and password managers use unique combinations for access, similar to rotating discs in the padlock problem to create secure passwords.

Genetic Algorithms:

- In data science, combinations are generated and optimized to find the best solutions, just like finding the best word in the padlock.

Supply Chain Optimization:

- Companies generate combinations of resources and schedules to maximize efficiency and reduce costs.

CONCLUSION

What We Learned:

- How combinations help in problem-solving.
- Importance of verification with a list of valid words.

Why It Matters:

- Simple techniques like these can solve real-world problems!