Concept Roadmap

Worlds

id	1
name	“World 1 - Hash Maps”
x_position	TBD
y_position	TBD
levels	- “Set Basics” - “Contains Duplicate” - “Missing Number” - “Find All Numbers Disappeared in an Array” - “Valid Sudoku” - “Set Conversion” - “Jewels and Stones” - “Set Operations” - “Intersection of Two Arrays” - “Map Basics” - “Map Counting” - “Majority Element” - “Majority Element II” - “Top K Frequent Elements” - “Map Comparison” - “Valid Anagram” - “Intersection of Two Arrays II” - “Valid Anagram” - “Ransom Note” - “String Mapping” - “Isomorphic Strings” - “Word Pattern” - “Group Anagrams” - “Index Mapping” - “Contains Duplicate II” - “Two Sum” - “Preset Mapping” - “Roman to Integer” - “Integer to Roman” - “Integer to English Words”

user_world	TBD
prerequisites	None
requiredBy	- “World 2 - Stack” - “World 3 - Two Pointers”
flip_arrow	TBD

id	2
name	“World 2 - Stacks”
x_position	TBD
y_position	TBD
levels	- “Stack Basics” - “Valid Parentheses” - “Remove All Adjacent Duplicates in String” - “Minimum String Length After Removing Substrings” - “Backspace String Compare” - “Baseball Game” - “Evaluate Reverse Polish Notation” - “Asteroid Collision” - “Tracking Depth” - “Maximum Nesting Depth of the Parentheses” - “Crawler Log Folder” - “Stack Reversal” - “Reverse Prefix of Word” - “Minimum Add to Make Parentheses Valid” - “Minimum Remove to Make Valid Parentheses” - “Score of Parentheses” - “Decode String” - “Longest Valid Parentheses” - “Monotonic Stack” - “Next Greater Element I” - “Final Prices With a Special Discount in a Shop” - “Remove K Digits” - “Daily Temperatures” - “Maximum Width Ramp”

	- “Largest Rectangle in Histogram”
user_world	TBD
prerequisites	- “World 1 - Hash Maps”
requiredBy	None
flip_arrow	TBD

id	3
name	“World 3 - Two Pointers”
x_position	TBD
y_position	TBD
levels	- “Two Pointers Basics” - “Reverse String” - “Checking and Swapping” - “Reverse Vowels of a String” - “Reverse Words in a String” - “Reordering by Swapping” - “Sort Array by Parity” - “Sort Array by Parity II” - “Sort Colors” - “Move Zeros” - “Remove Element” - “Remove Duplicates from Sorted Array” - “Two Structures” - “Is Subsequence” - “Find the Index of the First Occurence in a String” - “Compare Version Numbers” - “Merge Sorted Array” - “Assign Cookies” - “Shortest Distance to a Character” - “Advanced Checking” - “Valid Palindrome” - “Two Sum II - Input Array is Sorted” - “3Sum” - “4Sum” - “Container With Most Water” - “Trapping Rain Water”

user_world	TBD
prerequisites	- “World 1 - Hash Maps”
requiredBy	- “World 4 - Sliding Window” - “World 5 - Binary Search”
flip_arrow	TBD

id	4
name	“World 4 - Sliding Window”
x_position	TBD
y_position	TBD
levels	- “Rolling” - “Maximum Average Subarray I - “Defuse the Bomb” - “Find K Closest Elements” - “Grumpy Bookstore Owner” - “Inching” - “Longest Nice Substring” - “Minimum Size Subarray Sum” - “Combination Shortcut” - “Subarray Product Less Than K” - “Rolling Hash” - “Substrings of Size Three with Distinct Characters” - “Repeated DNA Sequence” - “Find All Anagrams in a String” - “Permutation in String” - “Inching Hash” - “Longest Substring Without Repeating Characters” - “Fruit Into Baskets” - “Longest Repeating Character Replacement” - “Minimum Window Substring” - “Binary Subarrays With Sum” - “Subarrays with K Different Integers”
user_world	TBD
prerequisites	- “World 3 - Two Pointers”

requiredBy	None
flip_arrow	TBD

id	5
name	“World 5 - Binary Search”
x_position	TBD
y_position	TBD
levels	- “Search a Structure” - “Binary Search” - “Search a 2D Matrix” - “Search in Rotated Sorted Array” - “Search in Rotated Sorted Array II” - “Search an Answer Space” - “Guess Number Higher or Lower” - “Valid Perfect Square” - “Uncertainty” - “Search Insert Position” - “First Bad Version” - “Find First and Last Position of Element in Sorted Array” - “Find Minimum in Rotated Sorted Array” - “Find Peak Element” - “Koko Eating Bananas” - “Median of Two Sorted Arrays”
user_world	TBD
prerequisites	- “World 3 - Two Pointers”
requiredBy	None
flip_arrow	TBD

id	6
name	“World 6 - Linked Lists”
x_position	TBD
y_position	TBD
levels	- “Linked List Basics”

	- “Remove Linked List Elements” - “Remove Duplicates from Sorted List” - “Multiple Pointers” - “Reverse Linked List” - “Remove Duplicates from Sorted List II” - “Reverse Linked List II” - “Reverse Nodes in k-Group” - “Swap Nodes in Pairs” - “Odd Even Linked List” - “Partition List” - “Remove Nth Node From End of List” - “Rotate List” - “Slow and Fast Pointers” - “Middle of the Linked List” - “Palindrome Linked List” - “Linked List Cycle” - “Linked List Cycle II” - “Multiple Lists” - “Merge Two Sorted Lists” - “Intersection of Two Linked Lists” - “Merge k Sorted Lists” - “Reorder List”
user_world	TBD
prerequisites	- “World 3 - Two Pointers”
requiredBy	- “World 7 - Trees”
flip_arrow	TBD

id	7
name	“World 7 - Trees”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD
prerequisites	- “World 6 - Linked Lists”
requiredBy	- “World 8 - Graphs”
flip_arrow	TBD

id	8
name	“World 8 - Graphs”
x_position	TBD
y_position	TBD
levels	- “Graph Basics” - “Depth First Search” - “Number of Islands” - “Surround Regions” - “Max Area of Island” - “Clone Graph” - “Breadth First Search” - “Keys and Rooms” - “Rotting Oranges” - “Course Schedule” - “Course Schedule II” - “Word Ladder” - “Word Ladder II”
user_world	TBD
prerequisites	- “World 7 - Trees”
requiredBy	- “World 9 - Heaps” - “World 10 - Backtracking”
flip_arrow	TBD

id	9
name	“World 9 - Heaps”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD
prerequisites	- “World 8 - Graphs”
requiredBy	None
flip_arrow	TBD

id	10
name	“World 10 - Backtracking”
x_position	TBD

y_position	TBD
levels	- “Getting Familiar” - “Subsets” - “Subsets II” - “Permutations” - “Permutations II” - “Combination Sum” - “Combination Sum II” - “Combination Sum III” - “Combination Sum IV” - “Generate Parentheses” - “Unique Binary Search Trees” - “Unique Binary Search Trees II” - “Sudoku Solver” - “N-Queens”
user_world	TBD
prerequisites	- “World 8 - Graphs”
requiredBy	- “World 11 - Dynamic Programming”
flip_arrow	TBD

id	11
name	“World 11 - Dynamic Programming”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD
prerequisites	- “World 10 - Backtracking”
requiredBy	- “World A - Greedy”
flip_arrow	TBD

id	12
name	- “World A - Greedy”
x_position	TBD
y_position	TBD
levels	TBD

user_world	TBD
prerequisites	- “World 11 - Dynamic Programming”
requiredBy	- “World B - Intervals” - “World C - Bit Manipulation” - “World D - Math”
flip_arrow	TBD

id	13
name	“World B - Intervals”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD
prerequisites	- “World A - Greedy”
requiredBy	None
flip_arrow	TBD

id	14
name	“World C - Bit Manipulation”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD
prerequisites	- “World A - Greedy”
requiredBy	None
flip_arrow	TBD

id	15
name	“World D - Math”
x_position	TBD
y_position	TBD
levels	TBD
user_world	TBD

prerequisites	- “World A - Greedy”
requiredBy	None
flip_arrow	TBD

Arrays / Strings

***** WARNING *** - Under Development - *** WARNING *****

Hash Set / Hash Map

id	1
title	“Set Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	20
y_position	20
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	None
requiredBy	- “Contains Duplicate”
flip_arrow	TBD

id	2
title	“Contains Duplicate”
description	“Given an integer array nums, return true if any value appears at least twice in the array, and return false if every element is distinct.”

type	PROBLEM
color	GREEN
name	“1”
x_position	20
y_position	120
leetcode_url	“https://leetcode.com/problems/contains-duplicate/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Set Basics”
requiredBy	- “Missing Number”
flip_arrow	TBD

id	3
title	“Missing Number”
description	“Given an array nums containing n distinct numbers in the range [0, n], return the only number in the range that is missing from the array.”
type	PROBLEM
color	GREEN
name	“2”
x_position	120
y_position	120
leetcode_url	“https://leetcode.com/problems/missing-number/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Contains Duplicate”
requiredBy	- “Find All Numbers Disappeared in an Array” - “Set Operations”
flip_arrow	TBD

id	4
title	“Find All Numbers Disappeared in an Array”
description	“Given an array nums of n integers where nums[i] is in the range [1, n], return an array of all the integers in the range [1, n] that do not appear in nums.”
type	PROBLEM
color	GREEN
name	“3”
x_position	220
y_position	120
leetcode_url	“https://leetcode.com/problems/find-all-numbers-disappeared-in-an-array/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Missing Number”
requiredBy	- “Valid Sudoku” - “Set Conversion” - “Set Operations”
flip_arrow	TBD

id	5
title	“Valid Sudoku”
description	“Given a 9×9 Sudoku board, determine if it is valid. A valid sudoku board must have the digits 1 through 9 in each row, column, and 3×3 box without repetition.”
type	BONUS
color	YELLOW
name	“A”
x_position	220
y_position	20
leetcode_url	“https://leetcode.com/problems/vali

	d-sudoku/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Find All Numbers Disappeared in an Array”
requiredBy	None
flip_arrow	TBD

id	6
title	“Set Conversion”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	320
y_position	120
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Find All Numbers Disappeared in an Array”
requiredBy	- “Jewels and Stones”
flip_arrow	TBD

id	7
title	“Jewels and Stones”
description	“Given 2 strings, jewels and stones, determine how many of the stones you have are also jewels.”
type	PROBLEM
color	GREEN
name	“4”
x_position	420
y_position	120
leetcode_url	“https://leetcode.com/problems/jewe

	ls-and-stones/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Set Conversion”
requiredBy	None
flip_arrow	TBD

id	8
title	“Set Operations”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	220
y_position	220
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	“Find All Numbers Disappeared in an Array”
requiredBy	- “Intersection of Two Arrays”
flip_arrow	TBD

id	9
title	“Intersection of Two Arrays”
description	“Given two integer arrays nums1 and nums2, return an array of their intersection, the distinct elements that appear in both arrays.”
type	PROBLEM
color	GREEN
name	“5”
x_position	320
y_position	220
leetcode_url	“https://leetcode.com/problems/inte

	rsection-of-two-arrays/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Set Operations”
requiredBy	- “Map Basics”
flip_arrow	TBD

id	10
title	“Map Basics”
description	None
type	LEARN
color	BLUE
name	“IV”
x_position	320
y_position	320
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Intersection of Two Arrays”
requiredBy	- “Map Counting”
flip_arrow	TBD

id	11
title	“Map Counting”
description	None
type	LEARN
color	BLUE
name	“V”
x_position	320
y_position	420
leetcode_url	None
world	“World 1 - Hash Maps”

world_id	1
prerequisites	- “Map Basics”
requiredBy	- “Majority Element”
flip_arrow	TBD

id	12
title	“Majority Element”
description	“Given an array nums of size n, return the majority element.”
type	PROBLEM
color	GREEN
name	“6”
x_position	420
y_position	420
leetcode_url	“https://leetcode.com/problems/majority-element/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Map Counting”
requiredBy	- “Majority Element II” - “Map Comparison”
flip_arrow	TBD

id	13
title	“Majority Element II”
description	“Given an array nums of size n, return all elements that appear at least a third of the time.”
type	PROBLEM
color	YELLOW
name	“7”
x_position	520
y_position	420
leetcode_url	“https://leetcode.com/problems/majority-element-ii/description/”

world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Majority Element”
requiredBy	- “Top K Frequent Elements”
flip_arrow	TBD

id	14
title	“Top K Frequent Elements”
description	“Given an integer array nums and an integer k, return the k most frequent elements.”
type	BONUS
color	YELLOW
name	“B”
x_position	620
y_position	420
leetcode_url	“https://leetcode.com/problems/top-k-frequent-elements/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Majority Element II”
requiredBy	None
flip_arrow	TBD

id	15
title	“Map Comparison”
description	None
type	LEARN
color	BLUE
name	“VI”
x_position	420
y_position	520
leetcode_url	None
world	“World 1 - Hash Maps”

world_id	1
prerequisites	- “Majority Element”
requiredBy	- “Valid Anagram”
flip_arrow	TBD

id	16
title	“Valid Anagram”
description	“Given two strings s and t, return true if t is an anagram of s, and false otherwise.”
type	PROBLEM
color	GREEN
name	“8”
x_position	420
y_position	620
leetcode_url	“https://leetcode.com/problems/valid-anagram/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Map Comparison”
requiredBy	- “Intersection of Two Arrays II” - “Ransom Note”
flip_arrow	TBD

id	17
title	“Intersection of Two Arrays II”
description	“Given two integer arrays nums1 and nums2, return an array of their intersection, ALL the elements that appear in both arrays.”
type	PROBLEM
color	GREEN
name	“9”
x_position	320

y_position	620
leetcode_url	“https://leetcode.com/problems/intersection-of-two-arrays-ii/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Valid Anagram”
requiredBy	None
flip_arrow	TBD

id	18
title	“Ransom Note”
description	“Given two strings ransomNote and magazine, return true if ransomNote can be constructed by using the letters from magazine and false otherwise.”
type	PROBLEM
color	GREEN
name	“10”
x_position	420
y_position	720
leetcode_url	“https://leetcode.com/problems/ransom-note/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Valid Anagram”
requiredBy	- “String Mapping”
flip_arrow	TBD

id	19
title	“String Mapping”
description	None
type	LEARN
color	BLUE

name	“VII”
x_position	520
y_position	720
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Ransom Note”
requiredBy	- “Isomorphic Strings”
flip_arrow	TBD

id	20
title	“Isomorphic Strings”
description	“Given two strings s and t, determine if they are isomorphic. Two strings s and t are isomorphic if the characters in s can be replaced to get t.“
type	PROBLEM
color	GREEN
name	“11”
x_position	620
y_position	720
leetcode_url	“https://leetcode.com/problems/isomorphic-strings/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “String Mapping”
requiredBy	- “Word Pattern” - “Index Mapping”
flip_arrow	TBD

id	21
title	“Word Pattern”
description	“Given a pattern and a string s, find if s follows the same pattern.

	s follows the pattern if the characters in pattern can be replaced with words to get s.”
type	PROBLEM
color	GREEN
name	“12”
x_position	620
y_position	820
leetcode_url	“https://leetcode.com/problems/word-pattern/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Isomorphic Strings”
requiredBy	- “Group Anagrams”
flip_arrow	TBD

id	22
title	“Group Anagrams”
description	“Given an array of strings strs, group the anagrams together.“
type	BONUS
color	YELLOW
name	“C”
x_position	620
y_position	920
leetcode_url	“https://leetcode.com/problems/group-anagrams/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Word Pattern”
requiredBy	None
flip_arrow	TBD

id	23
title	“Index Mapping”

description	None
type	LEARN
color	BLUE
name	“VIII”
x_position	720
y_position	720
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Isomorphic Strings”
requiredBy	- “Contains Duplicate II”
flip_arrow	TBD

id	24
title	“Contains Duplicate II”
description	“Given an integer array nums and an integer k, return true if any value appears at least twice in the array within k indices from each other, and false otherwise.”
type	PROBLEM
color	GREEN
name	“13”
x_position	820
y_position	720
leetcode_url	“https://leetcode.com/problems/contains-duplicate-ii/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Index Mapping”
requiredBy	- “Two Sum”
flip_arrow	TBD

id	25
title	“Two Sum”

description	“Given an array of integers nums and an integer target, return the indices of two numbers that add up to target.”
type	PROBLEM
color	GREEN
name	“14”
x_position	920
y_position	720
leetcode_url	“https://leetcode.com/problems/two-sum/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Contains Duplicate II”
requiredBy	- “Preset Mapping”
flip_arrow	TBD

id	26
title	“Preset Mapping”
description	None
type	LEARN
color	BLUE
name	“IX”
x_position	920
y_position	820
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Two Sum”
requiredBy	- “Roman to Integer”
flip_arrow	TBD

id	27
title	“Roman to Integer”
description	“Given a roman numeral, convert it

	to an integer.”
type	PROBLEM
color	GREEN
name	“15”
x_position	920
y_position	920
leetcode_url	“https://leetcode.com/problems/roman-to-integer/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Preset Mapping”
requiredBy	- “Integer to Roman” - “Integer to English Words”
flip_arrow	TBD

id	28
title	“Integer to Roman”
description	“Given an integer, convert it to a roman numeral.”
type	BONUS
color	YELLOW
name	“D”
x_position	1020
y_position	920
leetcode_url	“https://leetcode.com/problems/integer-to-roman/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Roman to Integer”
requiredBy	None
flip_arrow	TBD

id	29
title	“Integer to English Words”
description	“Convert a non-negative integer num

	to its English words representation.”
type	BONUS
color	RED
name	“E”
x_position	920
y_position	1020
leetcode_url	“https://leetcode.com/problems/integer-to-english-words/description/”
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Roman to Integer”
requiredBy	- “A Message”
flip_arrow	TBD

id	30
title	“A Message”
description	None
type	LEARN
color	BLUE
name	“X”
x_position	1020
y_position	1020
leetcode_url	None
world	“World 1 - Hash Maps”
world_id	1
prerequisites	- “Integer to English Words”
requiredBy	None
flip_arrow	TBD

Stack

id	1
title	“Stack Basics”
description	None

type	LEARN
color	BLUE
name	“I”
x_position	20
y_position	20
leetcode_url	None
world	“World 2 - Stacks”
world_id	2
prerequisites	None
requiredBy	- “Valid Parentheses”
flip_arrow	TBD

id	2
title	“Valid Parentheses”
description	“Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid.”
type	PROBLEM
color	GREEN
name	“1”
x_position	20
y_position	120
leetcode_url	None
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Stack Basics”
requiredBy	- “Remove All Adjacent Duplicates In String”
flip_arrow	TBD

id	3
title	“Remove All Adjacent Duplicates In String”
description	“Return a string after all

	duplicate removals have been made.”
type	PROBLEM
color	GREEN
name	“2”
x_position	20
y_position	220
leetcode_url	“https://leetcode.com/problems/remove-all-adjacent-duplicates-in-string/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Valid Parentheses”
requiredBy	- “Minimum String Length After Removing Substrings” - “Backspace String Compare”
flip_arrow	TBD

id	4
title	“Minimum String Length After Removing Substrings”
description	“Return the minimum possible length of the resulting string that you can obtain after removing instances of AB and CD.”
type	PROBLEM
color	GREEN
name	“3”
x_position	20
y_position	320
leetcode_url	“https://leetcode.com/problems/minimum-string-length-after-removing-substrings/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Remove All Adjacent Duplicates In String”
requiredBy	None

flip_arrow	TBD

id	5
title	“Backspace String Compare”
description	“Given two strings s and t, return true if they are equal when both are typed into empty text editors. '#' means a backspace character.”
type	PROBLEM
color	GREEN
name	“4”
x_position	120
y_position	220
leetcode_url	“https://leetcode.com/problems/backspace-string-compare/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Remove All Adjacent Duplicates In String”
requiredBy	- “Baseball Game” - “Tracking Depth”
flip_arrow	TBD

id	6
title	“Baseball Game”
description	“Return the sum of all the scores on the record after applying all the baseball game operations.”
type	PROBLEM
color	GREEN
name	“5”
x_position	220
y_position	220
leetcode_url	“https://leetcode.com/problems/baseball-game/description/”
world	“World 2 - Stacks”

world_id	2
prerequisites	- “Backspace String Compare”
requiredBy	- “Evaluate Reverse Polish Notation” - “Asteroid Collision”
flip_arrow	TBD

id	7
title	“Evaluate Reverse Polish Notation”
description	“Given an expression in reverse polish notation, evaluate the expression. Return an integer that represents the value of the expression.”
type	BONUS
color	YELLOW
name	“A”
x_position	220
y_position	120
leetcode_url	“https://leetcode.com/problems/evaluate-reverse-polish-notation/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Baseball Game”
requiredBy	None
flip_arrow	TBD

id	8
title	“Asteroid Collision”
description	“Find out the state of the asteroids after all collisions. If two asteroids meet, the smaller one will explode. If both are the same size, both will explode. Two asteroids moving in the same direction will never meet.”

type	BONUS
color	YELLOW
name	“B”
x_position	320
y_position	220
leetcode_url	“https://leetcode.com/problems/asteroid-collision/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Baseball Game”
requiredBy	None
flip_arrow	TBD

id	9
title	“Tracking Depth”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	120
y_position	320
leetcode_url	None
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Backspace String Compare”
requiredBy	- “Maximum Nesting Depth of the Parentheses”
flip_arrow	TBD

id	10
title	“Maximum Nesting Depth of the Parentheses”
description	“Given a valid parentheses string s, return the nesting depth of s. The nesting depth is the maximum

	number of nested parentheses.”
type	PROBLEM
color	GREEN
name	“6”
x_position	220
y_position	320
leetcode_url	“https://leetcode.com/problems/maximum-nesting-depth-of-the-parentheses/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Tracking Depth”
requiredBy	- “Crawler Log Folder” - “Minimum Add to Make Parentheses Valid”
flip_arrow	TBD

id	11
title	“Crawler Log Folder”
description	“Given a list of strings logs, return the minimum number of operations needed to go back to the main folder.”
type	PROBLEM
color	GREEN
name	“7”
x_position	120
y_position	420
leetcode_url	“https://leetcode.com/problems/crawler-log-folder/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Maximum Nesting Depth of the Parentheses”
requiredBy	- “Stack Reversal”
flip_arrow	TBD

id	12
title	“Stack Reversal”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	120
y_position	520
leetcode_url	None
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Crawler Log Folder”
requiredBy	- “Reverse Prefix of Word”
flip_arrow	TBD

id	13
title	“Reverse Prefix of Word”
description	“Given a 0-indexed string word and a character ch, reverse the segment of word that starts at index 0 and ends at the index of the first occurrence of ch”
type	PROBLEM
color	GREEN
name	“8”
x_position	120
y_position	620
leetcode_url	“https://leetcode.com/problems/reverse-prefix-of-word/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Stack Reversal”
requiredBy	None
flip_arrow	TBD

id	14
title	“Minimum Add to Make Parentheses Valid”
description	“You are given a parentheses string s. In one move, you can insert a parenthesis at any position of the string. Return the minimum number of moves required to make s valid.”
type	PROBLEM
color	YELLOW
name	“9”
x_position	320
y_position	420
leetcode_url	“https://leetcode.com/problems/minimum-add-to-make-parentheses-valid/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Maximum Nesting Depth of the Parentheses”
requiredBy	- “Minimum Remove to Make Valid Parentheses” - “Score of Parentheses”
flip_arrow	TBD

id	15
title	“Minimum Remove to Make Valid Parentheses”
description	“Given a string s of '(' , ')' and lowercase English characters. Your task is to remove the minimum number of parentheses ('(' or ')', in any positions) so that the resulting parentheses string is valid and return any valid string. ”
type	PROBLEM
color	YELLOW

name	“10”
x_position	320
y_position	520
leetcode_url	“https://leetcode.com/problems/minimum-remove-to-make-valid-parentheses/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Minimum Add to Make Parentheses Valid”
requiredBy	None
flip_arrow	TBD

id	16
title	“Score of Parentheses”
description	“Given a balanced parentheses string s, return the score of the string.”
type	PROBLEM
color	YELLOW
name	“11”
x_position	420
y_position	420
leetcode_url	“https://leetcode.com/problems/score-of-parentheses/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Minimum Add to Make Parentheses Valid”
requiredBy	- “Decode String - “Longest Valid Parentheses” - “Monotonic Stack”
flip_arrow	TBD

id	17
title	“Decode String”

description	“Given an encoded string, return its decoded string.”
type	BONUS
color	YELLOW
name	“C”
x_position	520
y_position	420
leetcode_url	“https://leetcode.com/problems/decode-string/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Score of Parentheses”
requiredBy	None
flip_arrow	TBD

id	18
title	“Longest Valid Parentheses”
description	“Given a string containing just the characters '(' and ')', return the length of the longest valid (well-formed) parentheses substring.”
type	BONUS
color	RED
name	“D”
x_position	420
y_position	520
leetcode_url	“https://leetcode.com/problems/longest-valid-parentheses/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Score of Parentheses”
requiredBy	None
flip_arrow	TBD

id	19
title	“Monotonic Stack”
description	None
type	LEARN
color	BLUE
name	“IV”
x_position	420
y_position	20
leetcode_url	None
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Score of Parentheses”
requiredBy	- “Next Greater Element”
flip_arrow	TBD

id	20
title	“Next Greater Element I”
description	“The next greater element of some element x in an array is the first greater element that is to the right of x in the same array. Return an array ans of length nums1.length such that ans[i] is the next greater element as described above.“
type	PROBLEM
color	GREEN
name	“12”
x_position	620
y_position	20
leetcode_url	“https://leetcode.com/problems/next-greater-element-i/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Monotonic Stack”

requiredBy	- “Final Prices With a Special Discount in a Shop” - “Remove K Digits”
flip_arrow	TBD

id	21
title	“Final Prices With a Special Discount in a Shop”
description	“You are given an integer array prices where prices[i] is the price of the ith item in a shop. Return an integer array answer where answer[i] is the final price you will pay for the ith item of the shop, considering a special discount.“
type	PROBLEM
color	GREEN
name	“13”
x_position	720
y_position	20
leetcode_url	“https://leetcode.com/problems/final-prices-with-a-special-discount-in-a-shop/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Next Greater Element I”
requiredBy	- “Daily Temperatures”
flip_arrow	TBD

id	22
title	“Remove K Digits”
description	“Given string num representing a non-negative integer num, and an integer k, return the smallest possible integer after removing k

	digits from num.”
type	PROBLEM
color	YELLOW
name	“14”
x_position	620
y_position	120
leetcode_url	“https://leetcode.com/problems/remove-k-digits/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Next Greater Element I”
requiredBy	- “Daily Temperatures”
flip_arrow	TBD

id	23
title	“Daily Temperatures”
description	“Given an array of integers temperatures, return an array answer such that answer[i] is the number of days you have to wait after the ith day to get a warmer temperature.”
type	PROBLEM
color	YELLOW
name	“15”
x_position	720
y_position	120
leetcode_url	“https://leetcode.com/problems/daily-temperatures/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Final Prices With a Special Discount in a Shop” - “Remove K Digits”
requiredBy	- “Maximum Width Ramp” - “Largest Rectangle in Histogram”

flip_arrow	TBD

id	24
title	“Maximum Width Ramp”
description	“A ramp in an integer array nums is a pair (i, j) for which i < j and nums[i] ⩽ nums[j]. The width of such a ramp is j - i. Given an integer array nums, return the maximum width of a ramp in nums.”
type	BONUS
color	YELLOW
name	“E”
x_position	820
y_position	120
leetcode_url	“https://leetcode.com/problems/maximum-width-ramp/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Daily Temperatures”
requiredBy	None
flip_arrow	TBD

id	25
title	“Largest Rectangle in Histogram”
description	“Given an array of integers heights representing the histogram's bar height where the width of each bar is 1, return the area of the largest rectangle in the histogram.”
type	BONUS
color	RED
name	“F”

x_position	720
y_position	220
leetcode_url	“https://leetcode.com/problems/maximum-width-ramp/description/”
world	“World 2 - Stacks”
world_id	2
prerequisites	- “Daily Temperatures”
requiredBy	None
flip_arrow	TBD

Two Pointers

id	1
title	“Two Pointers Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	20
y_position	20
leetcode_url	None
world	“World 3 - Two Pointers”
world_id	3
prerequisites	None
requiredBy	- “Reverse String”
flip_arrow	TBD

id	2
title	“Reverse String”
description	“Write a function that reverses a string. The input string is given as an array of characters s.”
type	PROBLEM
color	GREEN

name	“1”
x_position	20
y_position	120
leetcode_url	“https://leetcode.com/problems/reverse-string/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Two Pointers Basics”
requiredBy	- “Checking and Swapping”
flip_arrow	TBD

id	3
title	“Checking and Swapping”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	20
y_position	220
leetcode_url	None
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Reverse String”
requiredBy	- “Reverse Vowels of a String”
flip_arrow	TBD

id	4
title	“Reverse Vowels of a String”
description	“Given a string s, reverse only all the vowels in the string and return it.”
type	PROBLEM
color	GREEN
name	“2”

x_position	20
y_position	320
leetcode_url	“https://leetcode.com/problems/reverse-vowels-of-a-string/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Checking and Swapping”
requiredBy	- “Reverse Words in a String”
flip_arrow	TBD

id	5
title	“Reverse Words in a String”
description	“Given an input string s, reverse the order of the words.”
type	BONUS
color	YELLOW
name	“A”
x_position	20
y_position	420
leetcode_url	“https://leetcode.com/problems/reverse-words-in-a-string/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Reverse Vowels of a String”
requiredBy	None
flip_arrow	TBD

id	6
title	“Reordering by Swapping”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	120

y_position	220
leetcode_url	None
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Reverse Vowels of a String”
requiredBy	- “Sort Array by Parity”
flip_arrow	TBD

id	7
title	“Sort Array by Parity”
description	“Given an integer array nums, move all the even integers at the beginning of the array followed by all the odd integers.”
type	PROBLEM
color	GREEN
name	“3”
x_position	120
y_position	320
leetcode_url	“https://leetcode.com/problems/sort-array-by-parity/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Reordering by Swapping”
requiredBy	- “Sort Array by Parity II”
flip_arrow	TBD

id	8
title	“Sort Array by Parity II”
description	“Given an integer array nums, move all the even integers to even indices and odd integers to odd indices.”
type	PROBLEM
color	GREEN

name	“4”
x_position	120
y_position	420
leetcode_url	“https://leetcode.com/problems/sort-array-by-parity-ii/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Sort Array by Parity”
requiredBy	- “Sort Colors” - “Move Zeros”
flip_arrow	TBD

id	9
title	“Sort Colors”
description	“Given an array nums with n objects colored red, white, or blue, sort them in-place so that objects of the same color are adjacent, with the colors in the order red, white, and blue.”
type	BONUS
color	YELLOW
name	“B”
x_position	220
y_position	420
leetcode_url	“https://leetcode.com/problems/sort-colors/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Sort Array by Parity II”
requiredBy	None
flip_arrow	TBD

id	10
title	“Move Zeros”

description	“Given an integer array nums, move all 0's to the end of it while maintaining the relative order of the non-zero elements.”
type	PROBLEM
color	GREEN
name	“5”
x_position	120
y_position	520
leetcode_url	“https://leetcode.com/problems/sort-colors/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Sort Array by Parity II”
requiredBy	- “Remove Element”
flip_arrow	TBD

id	11
title	“Remove Element”
description	“Given an integer array nums and an integer val, remove all occurrences of val in nums in-place. The order of the elements may be changed. Then return the number of elements in nums which are not equal to val.”
type	PROBLEM
color	GREEN
name	“6”
x_position	220
y_position	520
leetcode_url	“https://leetcode.com/problems/remove-element/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Move Zeros”
requiredBy	- “Remove Duplicates from

	Sorted Array”
flip_arrow	TBD

id	12
title	“Remove Duplicates from Sorted Array”
description	“Given an integer array nums sorted in non-decreasing order, remove the duplicates in-place such that each unique element appears only once. The relative order of the elements should be kept the same. Then return the number of unique elements in nums.”
type	PROBLEM
color	GREEN
name	“7”
x_position	220
y_position	620
leetcode_url	“https://leetcode.com/problems/remove-duplicates-from-sorted-array/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Remove Element”
requiredBy	- “Two Structures” - “Advanced Checking”
flip_arrow	TBD

id	13
title	“Two Structures”
description	None
type	LEARN
color	BLUE
name	“IV”
x_position	320

y_position	620
leetcode_url	None
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Remove Duplicates from Sorted Array”
requiredBy	- “Is Subsequence”
flip_arrow	TBD

id	14
title	“Is Subsequence”
description	“Given two strings s and t, return true if s is a subsequence of t, or false otherwise.”
type	PROBLEM
color	GREEN
name	“8”
x_position	420
y_position	620
leetcode_url	“https://leetcode.com/problems/is-subsequence/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Two Structures”
requiredBy	- “Find the Index of the First Occurrence in a String” - “Merge Sorted Array”
flip_arrow	TBD

id	15
title	“Find the Index of the First Occurrence in a String”
description	“Given two strings needle and haystack, return the index of the first occurrence of needle in haystack, or -1 if needle is not

	part of haystack.”
type	PROBLEM
color	GREEN
name	“9”
x_position	520
y_position	620
leetcode_url	“https://leetcode.com/problems/find-the-index-of-the-first-occurrence-in-a-string/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Is Subsequence”
requiredBy	- “Compare Version Numbers”
flip_arrow	TBD

id	16
title	“Compare Version Number”
description	“Given two version strings, version1 and version2, compare them. A version string consists of revisions separated by dots '.'. The value of the revision is its integer conversion ignoring leading zeros.”
type	PROBLEM
color	YELLOW
name	“10”
x_position	520
y_position	720
leetcode_url	“https://leetcode.com/problems/compare-version-numbers/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Find the Index of the First Occurrence in a String”
requiredBy	None
flip_arrow	TBD

id	17
title	“Merge Sorted Array”
description	“You are given two integer arrays nums1 and nums2, sorted in non-decreasing order, and two integers m and n, representing the number of elements in nums1 and nums2 respectively.”
type	PROBLEM
color	GREEN
name	“11”
x_position	420
y_position	720
leetcode_url	“https://leetcode.com/problems/merge-sorted-array/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Is Subsequence”
requiredBy	- “Assign Cookies”
flip_arrow	TBD

id	18
title	“Assign Cookies”
description	“Given an integer array g representing each child’s greed factor and integer array s representing the size of a cookie, maximize the number of your content children and output the maximum number.”
type	PROBLEM
color	GREEN
name	“12”
x_position	420
y_position	820
leetcode_url	“https://leetcode.com/problems/assign-cookies/description/”

world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Merge Sorted Array”
requiredBy	- “Shortest Distance to a Character”
flip_arrow	TBD

id	19
title	“Shortest Distance to a Character”
description	“Given a string s and a character c that occurs in s, return an array of integers answer where answer.length == s.length and answer[i] is the distance from index i to the closest occurrence of character c in s.”
type	PROBLEM
color	GREEN
name	“13”
x_position	520
y_position	820
leetcode_url	“https://leetcode.com/problems/shortest-distance-to-a-character/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Assign Cookies”
requiredBy	None
flip_arrow	TBD

id	20
title	“Advanced Checking”
description	None
type	LEARN
color	BLUE
name	“V”

x_position	220
y_position	720
leetcode_url	None
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Remove Duplicates from Sorted Array”
requiredBy	- “Valid Palindrome”
flip_arrow	TBD

id	21
title	“Valid Palindrome”
description	“Given a string s, return true if it is a palindrome, or false otherwise.”
type	PROBLEM
color	GREEN
name	“14”
x_position	220
y_position	820
leetcode_url	“https://leetcode.com/problems/valid-palindrome/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Advanced Checking”
requiredBy	- “Two Sum II - Input Array is Sorted”
flip_arrow	TBD

id	22
title	“Two Sum II - Input Array is Sorted”
description	“Given a 1-indexed array of integers numbers that is already sorted in non-decreasing order, find two numbers such that they add

	up to a specific target number.”
type	PROBLEM
color	YELLOW
name	“15”
x_position	220
y_position	920
leetcode_url	“https://leetcode.com/problems/two-sum-ii-input-array-is-sorted/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Valid Palindrome”
requiredBy	- “3Sum” - “Container With Most Water”
flip_arrow	TBD

id	23
title	“3Sum”
description	“Given an integer array nums, return all the triplets that sum to 0.”
type	PROBLEM
color	YELLOW
name	“16”
x_position	170
y_position	1020
leetcode_url	“https://leetcode.com/problems/3sum/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Two Sum II - Input Array is Sorted”
requiredBy	- “4Sum”
flip_arrow	TBD

id	24
title	“4Sum”
description	“Given an integer array nums, return all the quadruplets that sum to 0.”
type	BONUS
color	YELLOW
name	“C”
x_position	170
y_position	1120
leetcode_url	“https://leetcode.com/problems/4sum/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “3Sum”
requiredBy	None
flip_arrow	TBD

id	25
title	“Container With Most Water”
description	“You are given an integer array height of length n. There are n vertical lines drawn such that the two endpoints of the ith line are (i, 0) and (i, height[i]). Find two lines that together with the x-axis form a container, such that the container contains the most water.”
type	PROBLEM
color	YELLOW
name	“17”
x_position	270
y_position	1020
leetcode_url	“https://leetcode.com/problems/container-with-most-water/description/”

world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Two Sum II - Input Array is Sorted”
requiredBy	- “Trapping Rain Water”
flip_arrow	TBD

id	26
title	“Trapping Rain Water”
description	“Given n non-negative integers representing an elevation map where the width of each bar is 1, compute how much water it can trap after raining.”
type	BONUS
color	RED
name	“D”
x_position	270
y_position	1120
leetcode_url	“https://leetcode.com/problems/trapping-rain-water/description/”
world	“World 3 - Two Pointers”
world_id	3
prerequisites	- “Container With Most Water”
requiredBy	None
flip_arrow	TBD

Sliding Window

id	1
title	“Rolling”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	250

y_position	50
leetcode_url	None
world	“World 4 - Sliding Window”
world_id	4
prerequisites	None
requiredBy	- “Maximum Average Subarray I”
flip_arrow	TBD

id	2
title	“Maximum Average Subarray I”
description	“You are given an integer array nums consisting of n elements, and an integer k. Find a contiguous subarray whose length is equal to k that has the maximum average value and return this value.”
type	PROBLEM
color	GREEN
name	“1”
x_position	250
y_position	150
leetcode_url	“https://leetcode.com/problems/maximum-average-subarray-i/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Rolling”
requiredBy	- “Defuse the Bomb”
flip_arrow	TBD

id	3
title	“Defuse the Bomb”
description	“Given the circular array code and an integer key k, return the decrypted code to defuse the bomb! The code for each index is the sum

	of the next k numbers.”
type	PROBLEM
color	GREEN
name	“2”
x_position	250
y_position	250
leetcode_url	“https://leetcode.com/problems/defuse-the-bomb/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Maximum Average Subarray I”
requiredBy	- “Find K Closest Elements” - “Grumpy Bookstore Owner” - “Inching”
flip_arrow	TBD

id	4
title	“Find K Closest Elements”
description	“Given a sorted integer array arr, two integers k and x, return the k closest integers to x in the array.”
type	BONUS
color	YELLOW
name	“A”
x_position	150
y_position	250
leetcode_url	“https://leetcode.com/problems/find-k-closest-elements/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Defuse the Bomb”
requiredBy	None
flip_arrow	TBD

id	5
title	“Grumpy Bookstore Owner”
description	“There is a bookstore owner that has a store open for n minutes. You are given an integer array customers of length n where customers[i] is the number of the customers that enter the store at the start of the ith minute and all those customers leave after the end of that minute. Return the maximum number of customers that can be satisfied throughout the day.”
type	BONUS
color	YELLOW
name	“B”
x_position	350
y_position	250
leetcode_url	“https://leetcode.com/problems/grumpy-bookstore-owner/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Defuse the Bomb”
requiredBy	None
flip_arrow	TBD

id	6
title	“Inching”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	250
y_position	350
leetcode_url	None
world	“World 4 - Sliding Window”

world_id	4
prerequisites	- “Defuse the Bomb”
requiredBy	- “Longest Nice String”
flip_arrow	TBD

id	7
title	“Longest Nice Substring”
description	“A string s is nice if, for every letter of the alphabet that s contains, it appears both in uppercase and lowercase. Given a string s, return the longest substring of s that is nice.”
type	PROBLEM
color	GREEN
name	“3”
x_position	250
y_position	450
leetcode_url	“https://leetcode.com/problems/longest-nice-substring/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Inching”
requiredBy	- “Best Time to Buy and Sell Stock” - “Minimum Size Subarray Sum”
flip_arrow	TBD

id	8
title	“Best Time to Buy and Sell Stock”
description	“You are given an array prices where prices[i] is the price of a given stock on the ith day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock.

	Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0.“
type	BONUS
color	GREEN
name	“C”
x_position	350
y_position	450
leetcode_url	“https://leetcode.com/problems/best-time-to-buy-and-sell-stock/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Longest Nice Substring”
requiredBy	None
flip_arrow	TBD

id	8
title	“Minimum Size Subarray Sum”
description	“Given an array of positive integers nums and a positive integer target, return the minimal length of a subarray whose sum is greater than or equal to target.”
type	PROBLEM
color	YELLOW
name	“4”
x_position	250
y_position	550
leetcode_url	“https://leetcode.com/problems/minimum-size-subarray-sum/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Longest Nice Substring”

requiredBy	- “Combination Shortcut”
flip_arrow	TBD

id	9
title	“Combination Shortcut”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	250
y_position	650
leetcode_url	None
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Minimum Size Subarray Sum”
requiredBy	- “Subarray Product Less Than K”
flip_arrow	TBD

id	10
title	“Subarray Product Less Than K”
description	“Given an array of integers nums and an integer k, return the number of contiguous subarrays where the product of all the elements in the subarray is strictly less than k.”
type	PROBLEM
color	YELLOW
name	“5”
x_position	250
y_position	750
leetcode_url	“https://leetcode.com/problems/subarray-product-less-than-k/description/”
world	“World 4 - Sliding Window”

world_id	4
prerequisites	- “Combination Shortcut”
requiredBy	- “Rolling Hash”
flip_arrow	TBD

id	11
title	“Rolling Hash”
description	None
type	LEARN
color	BLUE
name	“IV”
x_position	250
y_position	850
leetcode_url	None
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Subarray Product Less Than K”
requiredBy	- “Substrings of Size Three with Distinct Characters”
flip_arrow	TBD

id	12
title	“Substrings of Size Three with Distinct Characters”
description	“A string is good if there are no repeated characters. Given a string s, return the number of good substrings of length three in s.“
type	PROBLEM
color	GREEN
name	“6”
x_position	250
y_position	950
leetcode_url	“https://leetcode.com/problems/subs

	trings-of-size-three-with-distinct-characters/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Rolling Hash”
requiredBy	- “Repeated DNA Sequences”
flip_arrow	TBD

id	13
title	“Repeated DNA Sequences”
description	“Given a string s that represents a DNA sequence, return all the 10-letter-long sequences (substrings) that occur more than once in a DNA molecule. You may return the answer in any order.“
type	PROBLEM
color	YELLOW
name	“7”
x_position	250
y_position	1050
leetcode_url	“https://leetcode.com/problems/repeated-dna-sequences/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Substrings of Size Three with Distinct Characters”
requiredBy	- “Find All Anagrams in a String” - “Permutation in String”
flip_arrow	TBD

id	14
title	“Find All Anagrams in a String”
description	“Given two strings s and p, return an array of all the start indices

	of p's anagrams in s. You may return the answer in any order.“
type	PROBLEM
color	YELLOW
name	“8”
x_position	200
y_position	1150
leetcode_url	“https://leetcode.com/problems/find-all-anagrams-in-a-string/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Repeated DNA Sequences”
requiredBy	- “Inching Hash”
flip_arrow	TBD

id	15
title	“Permutation in String”
description	“Given two strings s1 and s2, return true if s2 contains a permutation of s1, or false otherwise.“
type	PROBLEM
color	YELLOW
name	“9”
x_position	300
y_position	1150
leetcode_url	“https://leetcode.com/problems/permutation-in-string/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Repeated DNA Sequences”
requiredBy	- “Inching Hash”
flip_arrow	TBD

id	16
title	“Inching Hash”
description	None
type	LEARN
color	BLUE
name	“V”
x_position	250
y_position	1250
leetcode_url	None
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Find All Anagrams in a String” - “Permutation in String”
requiredBy	- “Longest Substring Without Repeating Characters”
flip_arrow	TBD

id	17
title	“Longest Substring Without Repeating Characters”
description	“Given a string s, find the length of the longest substring without repeating characters.“
type	PROBLEM
color	YELLOW
name	“10”
x_position	250
y_position	1350
leetcode_url	“https://leetcode.com/problems/longest-substring-without-repeating-characters/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Inching Hash”
requiredBy	- “Fruit Into Baskets”

flip_arrow	TBD

id	18
title	“Fruit Into Baskets”
description	“Given the integer array fruits, return the maximum number of fruits you can pick.“
type	PROBLEM
color	YELLOW
name	“11”
x_position	250
y_position	1450
leetcode_url	“https://leetcode.com/problems/fruit-into-baskets/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Longest Substring Without Repeating Characters”
requiredBy	- “Longest Repeating Character Replacement” - “Binary Subarrays With Sum”
flip_arrow	TBD

id	19
title	“Longest Repeating Character Replacement”
description	“You are given a string s and an integer k. You can choose any character of the string and change it to any other uppercase English character. You can perform this operation at most k times. Return the length of the longest substring containing the same letter you can get after performing the above operations.“
type	PROBLEM

color	YELLOW
name	“12”
x_position	200
y_position	1550
leetcode_url	“https://leetcode.com/problems/longest-repeating-character-replacement/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Fruit Into Baskets”
requiredBy	- “Minimum Window Substring”
flip_arrow	TBD

id	20
title	“Minimum Window Substring”
description	“Given two strings s and t of lengths m and n respectively, return the minimum window substring of s such that every character in t (including duplicates) is included in the window.“
type	BONUS
color	RED
name	“C”
x_position	200
y_position	1650
leetcode_url	“https://leetcode.com/problems/minimum-window-substring/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Longest Repeating Character Replacement”
requiredBy	None
flip_arrow	TBD

id	21

title	“Binary Subarrays With Sum”
description	“Given a binary array nums and an integer goal, return the number of non-empty subarrays with a sum goal.“
type	PROBLEM
color	YELLOW
name	“13”
x_position	300
y_position	1550
leetcode_url	“https://leetcode.com/problems/binary-subarrays-with-sum/description/”
world	“World 4 - Sliding Window”
world_id	4
prerequisites	- “Fruit Into Baskets”
requiredBy	- “Subarrays with K Different Integers”
flip_arrow	TBD

id	22
title	“Subarrays with K Different Integers”
description	“Given an integer array nums and an integer k, return the number of good subarrays of nums. A good array is an array where the number of different integers in that array is exactly k.“
type	BONUS
color	RED
name	“D”
x_position	300
y_position	1650
leetcode_url	“https://leetcode.com/problems/subarrays-with-k-different-integers/description/”
world	“World 4 - Sliding Window”

world_id	4
prerequisites	- “Binary Subarrays With Sum”
requiredBy	None
flip_arrow	TBD

Binary Search

id	1
title	“Search a Structure”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	20
y_position	520
leetcode_url	None
world	“World 5 - Binary Search”
world_id	5
prerequisites	None
requiredBy	- “Binary Search”
flip_arrow	TBD

id	2
title	“Binary Search”
description	“Given an array of integers nums which is sorted in ascending order, and an integer target, write a function to search target in nums. If target exists, then return its index. Otherwise, return -1. You must write an algorithm with O(log n) runtime complexity.”
type	PROBLEM
color	GREEN
name	“1”
x_position	20

y_position	420
leetcode_url	“https://leetcode.com/problems/binary-search/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search a Structure”
requiredBy	- “Search a 2D Matrix” - “Search an Answer Space”
flip_arrow	TBD

id	3
title	“Search a 2D Matrix”
description	“Given an integer target, return true if target is in matrix or false otherwise. You must write a solution in O(log(m * n)) time complexity.”
type	PROBLEM
color	YELLOW
name	“2”
x_position	20
y_position	320
leetcode_url	“https://leetcode.com/problems/search-a-2d-matrix/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Binary Search”
requiredBy	- “Search in Rotated Sorted Array”
flip_arrow	TBD

id	4
title	“Search in Rotated Sorted Array”
description	“Given the sorted array nums after a possible rotation and an integer target, return the index of target

	if it is in nums, or -1 if it is not in nums. You must write an algorithm with O(log n) runtime complexity.”
type	PROBLEM
color	YELLOW
name	“3”
x_position	20
y_position	220
leetcode_url	“https://leetcode.com/problems/search-in-rotated-sorted-array/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search a 2D Matrix”
requiredBy	- “Search in Rotated Sorted Array II”
flip_arrow	TBD

id	5
title	“Search in Rotated Sorted Array II”
description	“Given the sorted array nums after a possible rotation and an integer target, return the index of target if it is in nums, or -1 if it is not in nums.You must decrease the overall operation steps as much as possible.”
type	BONUS
color	YELLOW
name	“A”
x_position	20
y_position	120
leetcode_url	“https://leetcode.com/problems/search-in-rotated-sorted-array-ii/description/”
world	“World 5 - Binary Search”

world_id	5
prerequisites	- “Search in Rotated Sorted Array”
requiredBy	None
flip_arrow	TBD

id	6
title	“Search an Answer Space”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	120
y_position	420
leetcode_url	None
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Binary Search”
requiredBy	- “Guess Number Higher or Lower”
flip_arrow	TBD

id	7
title	“Guess Number Higher or Lower”
description	“I pick a number from 1 to n. You have to guess which number I picked. Every time you guess wrong, I will tell you whether the number I picked is higher or lower than your guess. You call a pre-defined API int guess(int num), which returns whether you were higher, lower, or correct. Return the number that I picked.”
type	PROBLEM
color	GREEN

name	“4”
x_position	120
y_position	320
leetcode_url	“https://leetcode.com/problems/guess-number-higher-or-lower/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search an Answer Space”
requiredBy	- “Valid Perfect Square”
flip_arrow	TBD

id	8
title	“Valid Perfect Square”
description	“Given a positive integer num, return true if num is a perfect square or false otherwise. You must not use any built-in library function, such as sqrt.”
type	PROBLEM
color	GREEN
name	“5”
x_position	120
y_position	220
leetcode_url	“https://leetcode.com/problems/valid-perfect-square/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Guess Number Higher or Lower”
requiredBy	- “Uncertainty”
flip_arrow	TBD

id	8
title	“Uncertainty”

description	None
type	LEARN
color	BLUE
name	“III”
x_position	220
y_position	220
leetcode_url	None
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Valid Perfect Square”
requiredBy	- “Search Insert Position”
flip_arrow	TBD

id	9
title	“Search Insert Position”
description	“Given a sorted array of distinct integers and a target value, return the index if the target is found. If not, return the index where it would be if it were inserted in order. You must write an algorithm with O(log n) runtime complexity.“
type	PROBLEM
color	GREEN
name	“6”
x_position	220
y_position	120
leetcode_url	“https://leetcode.com/problems/search-insert-position/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Uncertainty”
requiredBy	- “First Bad Version” - “Find First and Last Position of Element in Sorted Array” - “Find Minimum in Rotated Sorted Array”

flip_arrow	TBD

id	10
title	“First Bad Version”
description	“Suppose you have n versions [1, 2, ... , n] and you want to find out the first bad one, which causes all the following ones to be bad. You are given an API bool isBadVersion(version) which returns whether version is bad. Implement a function to find the first bad version. You should minimize the number of calls to the API.“
type	PROBLEM
color	GREEN
name	“7”
x_position	120
y_position	120
leetcode_url	“https://leetcode.com/problems/first-bad-version/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search Insert Position”
requiredBy	None
flip_arrow	TBD

id	11
title	“Find First and Last Position of Element in Sorted Array”
description	“Given an array of integers nums sorted in non-decreasing order, find the starting and ending position of a given target value. You must write an algorithm with O(log n) runtime complexity.“
type	PROBLEM

color	YELLOW
name	“8”
x_position	320
y_position	120
leetcode_url	“https://leetcode.com/problems/find-first-and-last-position-of-element-in-sorted-array/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search Insert Position”
requiredBy	None
flip_arrow	TBD

id	12
title	“Find Minimum in Rotated Sorted Array”
description	“Given the sorted rotated array nums of unique elements, return the minimum element of this array. You must write an algorithm that runs in O(log n) time.“
type	PROBLEM
color	YELLOW
name	“9”
x_position	220
y_position	20
leetcode_url	“https://leetcode.com/problems/find-minimum-in-rotated-sorted-array/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Search Insert Position”
requiredBy	- “Find Peak Element” - “Koko Eating Bananas”
flip_arrow	TBD

id	13
title	“Find Peak Element”
description	“A peak element is an element that is strictly greater than its neighbors. Given a 0-indexed integer array nums, find a peak element, and return its index. If the array contains multiple peaks, return the index to any of the peaks.“
type	BONUS
color	YELLOW
name	“B”
x_position	220
y_position	-80
leetcode_url	“https://leetcode.com/problems/find-peak-element/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Find Minimum in Rotated Sorted Array”
requiredBy	None
flip_arrow	TBD

id	14
title	“Koko Eating Bananas”
description	“Koko loves to eat bananas. There are n piles of bananas, the ith pile has piles[i] bananas. The guards have gone and will come back in h hours. Koko can decide her bananas-per-hour eating speed of k. Each hour, she chooses some pile of bananas and eats k bananas from that pile. If the pile has less than k bananas, she eats all of

	them instead and will not eat any more bananas during this hour. Koko likes to eat slowly but still wants to finish eating all the bananas before the guards return. Return the minimum integer k such that she can eat all the bananas within h hours.“
type	PROBLEM
color	YELLOW
name	“10”
x_position	320
y_position	20
leetcode_url	“https://leetcode.com/problems/koko-eating-bananas/description/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Find Minimum in Rotated Sorted Array”
requiredBy	- “Median of Two Sorted Arrays”
flip_arrow	TBD

id	15
title	“Median of Two Sorted Arrays”
description	“Given two sorted arrays nums1 and nums2 of size m and n respectively, return the median of the two sorted arrays. The overall run time complexity should be O(log (m+n)).“
type	BONUS
color	RED
name	“C”
x_position	320
y_position	-80
leetcode_url	“https://leetcode.com/problems/median-of-two-sorted-arrays/description

	/”
world	“World 5 - Binary Search”
world_id	5
prerequisites	- “Koko Eating Bananas”
requiredBy	None
flip_arrow	TBD

Linked List

id	1
title	“Linked List Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	20
y_position	120
leetcode_url	None
world	“World 6 - Linked Lists”
world_id	6
prerequisites	None
requiredBy	- “Remove Linked List Elements”
flip_arrow	TBD

id	2
title	“Remove Linked List Elements”
description	“Given the head of a linked list and an integer val, remove all the nodes of the linked list that has Node.val == val, and return the new head.”
type	PROBLEM
color	GREEN
name	“1”
x_position	120

y_position	120
leetcode_url	“https://leetcode.com/problems/remove-linked-list-elements/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Linked List Basics”
requiredBy	- “Remove Duplicates from Sorted List”
flip_arrow	TBD

id	3
title	“Remove Duplicates from Sorted List”
description	“Given the head of a sorted linked list, delete all duplicates such that each element appears only once. Return the linked list sorted as well.”
type	PROBLEM
color	GREEN
name	“2”
x_position	220
y_position	120
leetcode_url	“https://leetcode.com/problems/remove-duplicates-from-sorted-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Remove Linked List Elements”
requiredBy	- “Multiple Pointers”
flip_arrow	TBD

id	4
title	“Multiple Pointers”
description	None

type	LEARN
color	BLUE
name	“II”
x_position	320
y_position	120
leetcode_url	None
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Remove Duplicates from Sorted List”
requiredBy	- “Reverse Linked List”
flip_arrow	TBD

id	5
title	“Reverse Linked List”
description	“Given the head of a singly linked list, reverse the list, and return the reversed list.”
type	PROBLEM
color	GREEN
name	“3”
x_position	420
y_position	120
leetcode_url	“https://leetcode.com/problems/reverse-linked-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Multiple Pointers”
requiredBy	- “Remove Duplicates from Sorted List” - “Reverse Linked List II” - “Swap Nodes in Pairs” - “Reorder List”
flip_arrow	TBD

id	6
title	“Remove Duplicates from Sorted List II”
description	“Given the head of a sorted linked list, delete all nodes that have duplicate numbers, leaving only distinct numbers from the original list. Return the linked list sorted as well.”
type	BONUS
color	YELLOW
name	“A”
x_position	520
y_position	20
leetcode_url	“https://leetcode.com/problems/remove-duplicates-from-sorted-list-ii/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Reverse Linked List”
requiredBy	None
flip_arrow	TBD

id	7
title	“Reverse Linked List II”
description	“Given the head of a singly linked list and two integers left and right where left ⩽ right, reverse the nodes of the list from position left to position right, and return the reversed list.”
type	PROBLEM
color	YELLOW
name	“4”
x_position	520
y_position	120
leetcode_url	“https://leetcode.com/problems/reve

	rse-linked-list-ii/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Reverse Linked List”
requiredBy	- “Reverse Nodes in k-Group”
flip_arrow	TBD

id	8
title	“Reverse Nodes in k-Group”
description	“Given the head of a linked list, reverse the nodes of the list k at a time, and return the modified list.”
type	BONUS
color	RED
name	“B”
x_position	620
y_position	120
leetcode_url	“https://leetcode.com/problems/reverse-nodes-in-k-group/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Reverse Linked List II”
requiredBy	None
flip_arrow	TBD

id	9
title	“Swap Nodes in Pairs”
description	“Given a linked list, swap every two adjacent nodes and return its head. You must solve the problem without modifying the values in the list's nodes (i.e., only nodes themselves may be changed.)”
type	PROBLEM
color	YELLOW

name	“5”
x_position	520
y_position	220
leetcode_url	“https://leetcode.com/problems/swap-nodes-in-pairs/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Reverse Linked List”
requiredBy	- “Odd Even Linked List” - “Remove Nth Node From End of List”
flip_arrow	TBD

id	10
title	“Odd Even Linked List”
description	“Given the head of a singly linked list, group all the nodes with odd indices together followed by the nodes with even indices, and return the reordered list.”
type	PROBLEM
color	YELLOW
name	“6”
x_position	620
y_position	220
leetcode_url	“https://leetcode.com/problems/odd-even-linked-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Swap Nodes in Pairs”
requiredBy	- “Partition List”
flip_arrow	TBD

id	11
title	“Partition List”

description	“Given the head of a linked list and a value x, partition it such that all nodes less than x come before nodes greater than or equal to x. You should preserve the original relative order of the nodes in each of the two partitions.”
type	BONUS
color	YELLOW
name	“C”
x_position	720
y_position	220
leetcode_url	“https://leetcode.com/problems/partition-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Odd Even Linked List”
requiredBy	None
flip_arrow	TBD

id	12
title	“Remove Nth Node From End of List”
description	“Given the head of a linked list and a value x, partition it such that all nodes less than x come before nodes greater than or equal to x. You should preserve the original relative order of the nodes in each of the two partitions.”
type	PROBLEM
color	YELLOW
name	“7”
x_position	620
y_position	320
leetcode_url	“https://leetcode.com/problems/remove-nth-node-from-end-of-list/descri

	ption/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Swap Nodes in Pairs”
requiredBy	- “Rotate List” - “Slow and Fast Pointers”
flip_arrow	TBD

id	13
title	“Rotate List”
description	“Given the head of a linked list, rotate the list to the right by k places.”
type	PROBLEM
color	YELLOW
name	“8”
x_position	720
y_position	320
leetcode_url	“https://leetcode.com/problems/rotate-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Remove Nth Node From End of List”
requiredBy	- “Multiple Lists”
flip_arrow	TBD

id	14
title	“Slow and Fast Pointers”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	720
y_position	420

leetcode_url	None
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Remove Nth Node From End of List”
requiredBy	- “Middle of the Linked List”
flip_arrow	TBD

id	15
title	“Middle of the Linked List”
description	“Given the head of a singly linked list, return the middle node of the linked list.”
type	PROBLEM
color	GREEN
name	“9”
x_position	820
y_position	420
leetcode_url	“https://leetcode.com/problems/middle-of-the-linked-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Slow and Fast Pointers”
requiredBy	- “Palindrome Linked List” - “Linked List Cycle” - “Reorder List”
flip_arrow	TBD

id	16
title	“Palindrome Linked List”
description	“Given the head of a singly linked list, return true if it is a palindrome or false otherwise.“
type	PROBLEM
color	GREEN

name	“10”
x_position	720
y_position	520
leetcode_url	“https://leetcode.com/problems/palindrome-linked-list/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Middle of the Linked List”
requiredBy	None
flip_arrow	TBD

id	17
title	“Linked List Cycle”
description	“Given head, the head of a linked list, determine if the linked list has a cycle in it.“
type	PROBLEM
color	GREEN
name	“11”
x_position	920
y_position	420
leetcode_url	“https://leetcode.com/problems/linked-list-cycle/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Middle of the Linked List”
requiredBy	- “Linked List Cycle II”
flip_arrow	TBD

id	18
title	“Linked List Cycle II”
description	“Given the head of a linked list, return the node where the cycle begins. If there is no cycle, return null.“

type	BONUS
color	YELLOW
name	“D”
x_position	920
y_position	520
leetcode_url	“https://leetcode.com/problems/linked-list-cycle/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Linked List Cycle”
requiredBy	- None
flip_arrow	TBD

id	19
title	“Multiple Lists”
description	None
type	LEARN
color	BLUE
name	“IV”
x_position	820
y_position	320
leetcode_url	None
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Rotate List”
requiredBy	- “Merge Two Sorted Lists”
flip_arrow	TBD

id	20
title	“Merge Two Sorted Lists”
description	“You are given the heads of two sorted linked lists list1 and list2. Merge the two lists into one sorted list. The list should be made by splicing together the nodes

	of the first two lists.“
type	PROBLEM
color	GREEN
name	“12”
x_position	1020
y_position	320
leetcode_url	“https://leetcode.com/problems/merge-two-sorted-lists/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Multiple Lists”
requiredBy	- “Intersection of Two Linked Lists” - “Merge k Sorted Lists” - “Reorder List”
flip_arrow	TBD

id	21
title	“Intersection of Two Linked Lists”
description	“Given the heads of two singly linked-lists headA and headB, return the node at which the two lists intersect. If the two linked lists have no intersection at all, return null.”
type	PROBLEM
color	GREEN
name	“13”
x_position	1020
y_position	220
leetcode_url	“https://leetcode.com/problems/intersection-of-two-linked-lists/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Merge Two Sorted Lists”
requiredBy	None

flip_arrow	TBD

id	22
title	“Merge k Sorted Lists”
description	“Given the heads of two singly linked-lists headA and headB, return the node at which the two lists intersect. If the two linked lists have no intersection at all, return null.”
type	BONUS
color	RED
name	“E”
x_position	1120
y_position	320
leetcode_url	“https://leetcode.com/problems/intersection-of-two-linked-lists/description/”
world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Merge Two Sorted Lists”
requiredBy	None
flip_arrow	TBD

id	23
title	“Reorder List”
description	“Given a linked list of length n, reorder the list so that the order is L1, Ln, L2, Ln-1, L3, Ln-2 …”
type	BONUS
color	YELLOW
name	“F”
x_position	920
y_position	1020
leetcode_url	“https://leetcode.com/problems/reorder-list/description/”

world	“World 6 - Linked Lists”
world_id	6
prerequisites	- “Middle of the Linked List” - “Reverse Linked List” - “Merge Two Sorted Lists”
requiredBy	None
flip_arrow	TBD

Trees

id	1
title	“Tree Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	TBD
y_position	TBD
leetcode_url	None
world	“World 7 - Trees”
world_id	7
prerequisites	None
requiredBy	- “Binary Tree Preorder Traversal” - “Binary Tree Inorder Traversal” - “Binary Tree Postorder Traversal”
flip_arrow	TBD

id	2
title	“Binary Tree Preorder Traversal”
description	“Given the root of a binary tree, return the preorder traversal of its nodes' values.”
type	PROBLEM

color	GREEN
name	“1”
x_position	TBD
y_position	TBD
leetcode_url	“https://leetcode.com/problems/binary-tree-preorder-traversal/description/”
world	“World 7 - Trees”
world_id	7
prerequisites	- “Tree Basics”
requiredBy	- “Invert Binary Tree”
flip_arrow	TBD

id	3
title	“Binary Tree Inorder Traversal”
description	“Given the root of a binary tree, return the inorder traversal of its nodes' values.”
type	PROBLEM
color	GREEN
name	“2”
x_position	TBD
y_position	TBD
leetcode_url	“https://leetcode.com/problems/binary-tree-inorder-traversal/description/”
world	“World 7 - Trees”
world_id	7
prerequisites	- “Tree Basics”
requiredBy	- “Invert Binary Tree”
flip_arrow	TBD

id	4
title	“Binary Tree Postorder Traversal”
description	“Given the root of a binary tree,

	return the postorder traversal of its nodes' values.”
type	PROBLEM
color	GREEN
name	“3”
x_position	TBD
y_position	TBD
leetcode_url	“https://leetcode.com/problems/binary-tree-postorder-traversal/description/”
world	“World 7 - Trees”
world_id	7
prerequisites	- “Tree Basics”
requiredBy	- “Invert Binary Tree”
flip_arrow	TBD

Graphs

id	1
title	“Graph Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	250
y_position	50
leetcode_url	None
world	“World 8 - Graphs”
world_id	8
prerequisites	None
requiredBy	- “Depth First Search” - “Breadth First Search”
flip_arrow	TBD

id	2

title	“Depth First Search”
description	None
type	LEARN
color	BLUE
name	“II”
x_position	200
y_position	150
leetcode_url	None
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Graph Basics”
requiredBy	- “Number of Islands” - “Keys and Rooms”
flip_arrow	TBD

id	3
title	“Number of Islands”
description	“Given an m x n 2D binary grid grid which represents a map of '1's (land) and '0's (water), return the number of islands.”
type	PROBLEM
color	YELLOW
name	“2”
x_position	150
y_position	250
leetcode_url	“https://leetcode.com/problems/number-of-islands/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Depth First Search”
requiredBy	- “Surround Regions”
flip_arrow	TBD

id	4

title	“Surround Regions”
description	“You are given an m x n matrix board containing letters 'X' and 'O', capture regions of ‘O’ that are surrounded by ‘X’. To capture a surrounded region, replace all 'O's with 'X's in-place within the original board.”
type	PROBLEM
color	YELLOW
name	“3”
x_position	50
y_position	250
leetcode_url	“https://leetcode.com/problems/surrounded-regions/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Number of Islands”
requiredBy	None
flip_arrow	TBD

id	5
title	“Max Area of Island”
description	“Given an m x n 2D binary grid grid which represents a map of '1's (land) and '0's (water), return the size of the largest island.”
type	PROBLEM
color	YELLOW
name	“4”
x_position	150
y_position	350
leetcode_url	“https://leetcode.com/problems/max-area-of-island/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Number of Islands”

requiredBy	- “Clone Graph”
flip_arrow	TBD

id	6
title	“Clone Graph”
description	“Given a reference of a node in a connected undirected graph, return a deep copy (clone) of the graph.”
type	BONUS
color	YELLOW
name	“A”
x_position	150
y_position	450
leetcode_url	“https://leetcode.com/problems/clone-graph/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Max Area of Island”
requiredBy	None
flip_arrow	TBD

id	7
title	“Breadth First Search”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	300
y_position	150
leetcode_url	None
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Graph Basics”
requiredBy	- “Keys and Rooms” - “Rotting Oranges”

flip_arrow	TBD

id	8
title	“Keys and Rooms”
description	“Given an array rooms where rooms[i] is the set of keys that you can obtain if you visited room i, return true if you can visit all the rooms, or false otherwise.”
type	PROBLEM
color	YELLOW
name	“1”
x_position	250
y_position	250
leetcode_url	“https://leetcode.com/problems/keys-and-rooms/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Depth First Search” - “Breadth First Search”
requiredBy	None
flip_arrow	TBD

id	8
title	“Rotting Oranges”
description	“Every minute, any fresh orange that is 4-directionally adjacent to a rotten orange becomes rotten. Return the minimum number of minutes that must elapse until no cell has a fresh orange. If this is impossible, return -1.”
type	PROBLEM
color	YELLOW
name	“5”
x_position	350

y_position	250
leetcode_url	“https://leetcode.com/problems/rotting-oranges/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Breadth First Search”
requiredBy	- “Course Schedule”
flip_arrow	TBD

id	9
title	“Course Schedule”
description	“There are a total of numCourses courses you have to take, labeled from 0 to numCourses - 1. You are given an array prerequisites where prerequisites[i] = [ai, bi] indicates that you must take course bi first if you want to take course ai. Return true if you can finish all courses. Otherwise, return false.”
type	PROBLEM
color	YELLOW
name	“6”
x_position	350
y_position	350
leetcode_url	“https://leetcode.com/problems/course-schedule/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Rotting Oranges”
requiredBy	- “Word Ladder” - “Course Schedule II”
flip_arrow	TBD

id	10

title	“Course Schedule II”
description	“There are a total of numCourses courses you have to take, labeled from 0 to numCourses - 1. You are given an array prerequisites where prerequisites[i] = [ai, bi] indicates that you must take course bi first if you want to take course ai. Return any valid order of classes that can be taken.”
type	BONUS
color	YELLOW
name	“B”
x_position	450
y_position	350
leetcode_url	“https://leetcode.com/problems/course-schedule/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Course Schedule”
requiredBy	None
flip_arrow	TBD

id	11
title	“Word Ladder”
description	“Given two words, beginWord and endWord, and a dictionary wordList, return the number of words in the shortest transformation sequence from beginWord to endWord, or 0 if no such sequence exists.”
type	BONUS
color	RED
name	“C”
x_position	350
y_position	450
leetcode_url	“https://leetcode.com/problems/word-ladder/description/”

world	“World 8 - Graphs”
world_id	8
prerequisites	- “Course Schedule”
requiredBy	- “Word Ladder II”
flip_arrow	TBD

id	11
title	“Word Ladder II”
description	“Given two words, beginWord and endWord, and a dictionary wordList, return ALL of the shortest transformation sequences from beginWord to endWord, or an empty list if no such sequence exists.”
type	BONUS
color	RED
name	“D”
x_position	450
y_position	450
leetcode_url	“https://leetcode.com/problems/word-ladder-ii/description/”
world	“World 8 - Graphs”
world_id	8
prerequisites	- “Word Ladder”
requiredBy	None
flip_arrow	TBD

Heaps

id	1
title	“Heap Basics”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	20

y_position	20
leetcode_url	None
world	“World 9 - Heaps”
world_id	9
prerequisites	None
requiredBy	- “Last Stone Weight”
flip_arrow	

id	2
title	“Last Stone Weight”
description	“You are given an array of integers stones where stones[i] is the weight of the ith stone. On each turn, choose the heaviest two stones and smash them together, leaving the difference in the weights as a new stone. At the end of the game, there is at most one stone left. Return the weight of the last remaining stone. If there are no stones left, return 0.“
type	PROBLEM
color	GREEN
name	“1”
x_position	120
y_position	120
leetcode_url	“https://leetcode.com/problems/last-stone-weight/description/”
world	“World 9 - Heaps”
world_id	9
prerequisites	- “Heap Basics”
requiredBy	- “Kth Largest Element in an Array”
flip_arrow	

id	3

title	“Kth Largest Element in an Array”
description	“Given an integer array nums and an integer k, return the kth largest element in the array without sorting the array.”
type	PROBLEM
color	YELLOW
name	“2”
x_position	20
y_position	220
leetcode_url	“https://leetcode.com/problems/kth-largest-element-in-an-array/description/”
world	“World 9 - Heaps”
world_id	9
prerequisites	- “Last Stone Weight”
requiredBy	- “Task Scheduler”
flip_arrow	

id	4
title	“K Closest Points to Origin”
description	“Given an array of points where points[i] = [xi, yi] represents a point on the X-Y plane and an integer k, return the k closest points to the origin (0, 0).”
type	PROBLEM
color	YELLOW
name	“3”
x_position	120
y_position	320
leetcode_url	“https://leetcode.com/problems/task-scheduler/description/”
world	“World 9 - Heaps”
world_id	9
prerequisites	- “Kth Largest Element in an Array”

requiredBy	- “Task Scheduler”
flip_arrow	

id	5
title	“Task Scheduler”
description	“You are given an array of CPU tasks and a number n. Each CPU interval can be idle or allow the completion of one task. Tasks can be completed in any order, but there has to be a gap of at least n intervals between two tasks with the same label. Return the minimum number of CPU intervals required to complete all tasks.”
type	PROBLEM
color	YELLOW
name	“4”
x_position	20
y_position	420
leetcode_url	“https://leetcode.com/problems/task-scheduler/description/”
world	“World 9 - Heaps”
world_id	9
prerequisites	- “K Closest Points to Origin”
requiredBy	- “Find Median From Data Stream” - “IPO”
flip_arrow	

id	6
title	“Find Median From Data Stream”
description	“The median is the middle value in an ordered integer list. If the size of the list is even, there is no middle value, and the median is the mean of the two middle values.

	Implement the MedianFinder class.”
type	BONUS
color	RED
name	“A”
x_position	-80
y_position	520
leetcode_url	“https://leetcode.com/problems/find-median-from-data-stream/description/”
world	“World 9 - Heaps”
world_id	9
prerequisites	- “Task Scheduler”
requiredBy	None
flip_arrow	

id	7
title	“IPO”
description	“You are given n projects where the ith project has a pure profit profits[i] and a minimum capital of capital[i] is needed to start it. Initially, you have w capital. When you finish a project, you will obtain its pure profit and the profit will be added to your total capital. Pick a list of at most k distinct projects from given projects to maximize your final capital, and return the final maximized capital.”
type	BONUS
color	RED
name	“B”
x_position	120
y_position	520
leetcode_url	“https://leetcode.com/problems/ipo/description/”
world	“World 9 - Heaps”

world_id	9
prerequisites	- “Task Scheduler”
requiredBy	None
flip_arrow	

Backtracking

id	1
title	“Getting Familiar”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	50
y_position	50
leetcode_url	None
world	“World 10 - Backtracking”
world_id	10
prerequisites	None
requiredBy	- “Subsets”
flip_arrow	TBD

id	2
title	“Subsets”
description	“Given an integer array nums of unique elements, return all possible subsets (the power set). The solution set must not contain duplicate subsets. Return the solution in any order.“
type	PROBLEM
color	YELLOW
name	“1”
x_position	150
y_position	150

leetcode_url	“https://leetcode.com/problems/subsets/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Getting Familiar”
requiredBy	- “Subsets II” - “Permutations”
flip_arrow	TBD

id	3
title	“Subsets II”
description	“Given an integer array nums of elements that may contain duplicates, return all possible subsets (the power set). The solution set must not contain duplicate subsets. Return the solution in any order.“
type	BONUS
color	YELLOW
name	“A”
x_position	250
y_position	50
leetcode_url	“https://leetcode.com/problems/subsets-ii/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Subsets”
requiredBy	None
flip_arrow	TBD

id	4
title	“Permutations”
description	“Given an integer array nums of unique elements, return all the possible

	permutations. You can return the answer in any order.“
type	PROBLEM
color	YELLOW
name	“2”
x_position	250
y_position	250
leetcode_url	“https://leetcode.com/problems/permutations/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Subsets”
requiredBy	- “Permutations II” - “Combination Sum”
flip_arrow	TBD

id	5
title	“Permutations II”
description	“Given an integer array nums of elements that may contain duplicates, return all the possible permutations. You can return the answer in any order.“
type	BONUS
color	YELLOW
name	“B”
x_position	350
y_position	150
leetcode_url	“https://leetcode.com/problems/permutations/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Permutations”
requiredBy	None
flip_arrow	TBD

id	6
title	“Combination Sum”
description	“Given an array of distinct integers candidates and a target integer target, return a list of all unique combinations of candidates where the chosen numbers sum to target. The numbers in candidates can be used an unlimited number of times in each combination.“
type	PROBLEM
color	YELLOW
name	“3”
x_position	350
y_position	350
leetcode_url	“https://leetcode.com/problems/combination-sum/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Permutations”
requiredBy	- “Combination Sum II” - “Generate Parentheses”
flip_arrow	TBD

id	7
title	“Combination Sum II”
description	“Given an array of distinct integers candidates and a target integer target, return a list of all unique combinations of candidates where the chosen numbers sum to target. The numbers in candidates can only be used once in each combination.“
type	BONUS
color	YELLOW

name	“C”
x_position	450
y_position	250
leetcode_url	“https://leetcode.com/problems/combination-sum-ii/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Combination Sum”
requiredBy	- “Combination Sum III”
flip_arrow	TBD

id	8
title	“Combination Sum III”
description	“Find all valid combinations of k numbers that sum up to n such that only numbers 1 through 9 are used and each number is used at most once.“
type	BONUS
color	YELLOW
name	“D”
x_position	550
y_position	150
leetcode_url	“https://leetcode.com/problems/combination-sum-iii/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Combination Sum II”
requiredBy	- “Combination Sum IV”
flip_arrow	TBD

id	9
title	“Combination Sum IV”
description	“Given an array of distinct integers nums and a target integer

	target, return the number of possible combinations that add up to target. In this case, difference sequences count as different combinations.“
type	BONUS
color	YELLOW
name	“E”
x_position	650
y_position	50
leetcode_url	“https://leetcode.com/problems/combination-sum-iv/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Combination Sum III”
requiredBy	None
flip_arrow	TBD

id	10
title	“Generate Parentheses”
description	“Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses.“
type	PROBLEM
color	YELLOW
name	“4”
x_position	450
y_position	450
leetcode_url	“https://leetcode.com/problems/generate-parentheses/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Combination Sum”
requiredBy	- “Unique Binary Search Trees”
flip_arrow	TBD

id	11
title	“Unique Binary Search Trees”
description	“Given an integer n, return the number of structurally unique BST's (binary search trees) which has exactly n nodes of unique values from 1 to n.“
type	PROBLEM
color	YELLOW
name	“5”
x_position	550
y_position	550
leetcode_url	“https://leetcode.com/problems/unique-binary-search-trees/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Generate Parentheses”
requiredBy	- “Unique Binary Search Trees II” - “Sudoku Solver” - “N-Queens”
flip_arrow	TBD

id	12
title	“Unique Binary Search Trees II”
description	“Given an integer n, return ALL the unique BST's (binary search trees) which has exactly n nodes of unique values from 1 to n.“
type	BONUS
color	YELLOW
name	“F”
x_position	650
y_position	450
leetcode_url	“https://leetcode.com/problems/unique-binary-search-trees-ii/descripti

	on/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Unique Binary Search Trees”
requiredBy	None
flip_arrow	TBD

id	13
title	“Sudoku Solver”
description	“Given a 9×9 Sudoku board, fill in the rest of the board such that it is valid. A valid sudoku board must have the digits 1 through 9 in each row, column, and 3×3 box without repetition.”
type	BONUS
color	RED
name	“G”
x_position	450
y_position	650
leetcode_url	“https://leetcode.com/problems/sudoku-solver/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Unique Binary Search Trees”
requiredBy	None
flip_arrow	TBD

id	14
title	“N-Queens”
description	“The n-queens puzzle is the problem of placing n queens on an n x n chessboard such that no two queens attack each other. Given an integer n, return all distinct solutions to the n-queens puzzle. You may return

	the answer in any order.”
type	PROBLEM
color	RED
name	“6”
x_position	650
y_position	650
leetcode_url	“https://leetcode.com/problems/n-queens/description/”
world	“World 10 - Backtracking”
world_id	10
prerequisites	- “Unique Binary Search Trees”
requiredBy	None
flip_arrow	TBD

Dynamic Programming

id	1
title	“Backtracking with Memory”
description	None
type	LEARN
color	BLUE
name	“I”
x_position	250
y_position	50
leetcode_url	None
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	None
requiredBy	- “Variable DP”
flip_arrow	TBD

id	2
title	“Variable DP”
description	None
type	LEARN

color	BLUE
name	“II”
x_position	150
y_position	150
leetcode_url	None
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Backtracking with Memory”
requiredBy	- “Climbing Stairs” - “Array DP”
flip_arrow	TBD

id	3
title	“Climbing Stairs”
description	“You are climbing a staircase. It takes n steps to reach the top. Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?“
type	PROBLEM
color	GREEN
name	“1”
x_position	50
y_position	250
leetcode_url	“https://leetcode.com/problems/climbing-stairs/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Variable DP”
requiredBy	- “Count Vowels Permutation”
flip_arrow	TBD

id	4
title	“Count Vowels Permutation”
description	“You are climbing a staircase. It

	takes n steps to reach the top. Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?“
type	BONUS
color	RED
name	“A”
x_position	50
y_position	350
leetcode_url	“https://leetcode.com/problems/count-vowels-permutation/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Climbing Stairs”
requiredBy	None
flip_arrow	TBD

id	5
title	“Array DP”
description	None
type	LEARN
color	BLUE
name	“III”
x_position	350
y_position	250
leetcode_url	None
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Variable DP”
requiredBy	- “Pascal’s Triangle” - “Min Cost Climbing Stairs”
flip_arrow	TBD

id	6
title	“Pascal’s Triangle”

description	“Given an integer numRows, return the first numRows of Pascal's triangle. In Pascal's triangle, each number is the sum of the two numbers directly above it.“
type	PROBLEM
color	GREEN
name	“2”
x_position	450
y_position	250
leetcode_url	“https://leetcode.com/problems/pascals-triangle/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Array DP”
requiredBy	None
flip_arrow	TBD

id	7
title	“Min Cost Climbing Stairs”
description	“You are given an integer array cost where cost[i] is the cost of ith step on a staircase. Once you pay the cost, you can either climb one or two steps. You can either start from the step with index 0, or the step with index 1. Return the minimum cost to reach the top of the floor.“
type	PROBLEM
color	GREEN
name	“3”
x_position	450
y_position	350
leetcode_url	“https://leetcode.com/problems/min-cost-climbing-stairs/description/”

world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Array DP”
requiredBy	- “House Robber”
flip_arrow	TBD

id	8
title	“House Robber”
description	“Given an integer array nums representing the amount of money of each house, return the maximum amount of money you can rob tonight without alerting the police. Police are alerted if two adjacent houses are robbed.“
type	PROBLEM
color	YELLOW
name	“4”
x_position	450
y_position	450
leetcode_url	“https://leetcode.com/problems/house-robber/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Min Cost Climbing Stairs”
requiredBy	- “House Robber II” - “Coin Change”
flip_arrow	TBD

id	8
title	“House Robber II”
description	“Given a circular integer array nums representing the amount of money of each house, return the maximum amount of money you can rob tonight without alerting the

	police. Police are alerted if two adjacent houses are robbed. The last house is considered adjacent to the first house.“
type	BONUS
color	YELLOW
name	“B”
x_position	550
y_position	450
leetcode_url	“https://leetcode.com/problems/house-robber-ii/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “House Robber”
requiredBy	None
flip_arrow	TBD

id	9
title	“Coin Change”
description	“You are given an integer array coins representing coins of different denominations and an integer amount representing a total amount of money. Return the fewest number of coins that you need to make up that amount. If that amount of money cannot be made up by any combination of the coins, return -1.“
type	PROBLEM
color	YELLOW
name	“5”
x_position	350
y_position	550
leetcode_url	“https://leetcode.com/problems/coin-change/description/”
world	“World 11 - Dynamic Programming”

world_id	11
prerequisites	- “House Robber”
requiredBy	- “Decode Ways”
flip_arrow	TBD

id	10
title	“Decode Ways”
description	“Given a string s containing only digits, return the number of ways to decode it into letters. If the entire string cannot be decoded in any valid way, return 0.“
type	PROBLEM
color	YELLOW
name	“6”
x_position	250
y_position	650
leetcode_url	“https://leetcode.com/problems/decode-ways/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Coin Change”
requiredBy	- “2D DP”
flip_arrow	TBD

id	11
title	“Word Break”
description	“Given a string s and a dictionary of strings wordDict, return true if s can be segmented into a space-separated sequence of one or more dictionary words.“
type	PROBLEM
color	YELLOW
name	“7”
x_position	150

y_position	650
leetcode_url	“https://leetcode.com/problems/word-break/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Decode Ways”
requiredBy	- “Word Break II”
flip_arrow	TBD

id	12
title	“Word Break II”
description	“Given a string s and a dictionary of strings wordDict, return ALL of the ways s can be segmented into a space-separated sequence of one or more dictionary words.“
type	BONUS
color	RED
name	“C”
x_position	50
y_position	650
leetcode_url	“https://leetcode.com/problems/word-break-ii/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Word Break”
requiredBy	None
flip_arrow	TBD

id	13
title	“2D DP”
description	None
type	LEARN
color	BLUE
name	“IV”

x_position	250
y_position	850
leetcode_url	None
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Decode Ways”
requiredBy	- “Longest Common Subsequence”
flip_arrow	TBD

id	14
title	“Longest Common Subsequence”
description	“Given two strings text1 and text2, return the length of their longest common subsequence. If there is no common subsequence, return 0.”
type	PROBLEM
color	YELLOW
name	“8”
x_position	250
y_position	950
leetcode_url	“https://leetcode.com/problems/longest-common-subsequence/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “2D DP”
requiredBy	- “Edit Distance” - “Coin Change II”
flip_arrow	TBD

id	15
title	“Edit Distance”
description	“Given two strings word1 and word2, return the minimum number of operations required to convert word1 to word2. The operations are

	deleting a character, inserting a character, or replacing a character.“
type	BONUS
color	YELLOW
name	“D”
x_position	350
y_position	950
leetcode_url	“https://leetcode.com/problems/edit-distance/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Longest Common Subsequence”
requiredBy	None
flip_arrow	TBD

id	16
title	“Coin Change II”
description	“You are given an integer array coins representing coins of different denominations and an integer amount representing a total amount of money. Return the number of combinations that make up that amount. If that amount of money cannot be made up by any combination of the coins, return 0.“
type	PROBLEM
color	YELLOW
name	“9”
x_position	250
y_position	1050
leetcode_url	“https://leetcode.com/problems/coin-change-ii/description/”
world	“World 11 - Dynamic Programming”
world_id	11

prerequisites	- “Longest Common Subsequence”
requiredBy	- “Interleaving String”
flip_arrow	TBD

id	17
title	“Interleaving String”
description	“Given strings s1, s2, and s3, find whether s3 is formed by an interleaving of s1 and s2.“
type	PROBLEM
color	YELLOW
name	“10”
x_position	250
y_position	1150
leetcode_url	“https://leetcode.com/problems/interleaving-string/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Coin Change II”
requiredBy	- “Distinct Subsequences” - “Regular Expression Matching” - “Wildcard Matching”
flip_arrow	TBD

id	18
title	“Distinct Subsequences”
description	“Given two strings s and t, return the number of distinct subsequences of s which equals t.“
type	BONUS
color	RED
name	“E”
x_position	150
y_position	1500
leetcode_url	“https://leetcode.com/problems/dist

	inct-subsequences/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Interleaving String”
requiredBy	- “Legacy Set”
flip_arrow	TBD

id	19
title	“Regular Expression Matching”
description	“Given an input string s and a pattern p, implement regular expression matching with support for '.' and '*', where '.' matches any single character and '*' matches zero or more of the preceding element.”
type	BONUS
color	RED
name	“F”
x_position	250
y_position	1500
leetcode_url	“https://leetcode.com/problems/regular-expression-matching/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Interleaving String”
requiredBy	- “Legacy Set”
flip_arrow	TBD

id	20
title	“Wildcard Matching”
description	“Given an input string s and a pattern p, implement wildcard matching with support for '?' and '*', where '?' matches any single

	character and '*' matches any sequence of characters.”
type	BONUS
color	RED
name	“G”
x_position	350
y_position	1500
leetcode_url	“https://leetcode.com/problems/wildcard-matching/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Interleaving String”
requiredBy	- “Legacy Set”
flip_arrow	TBD

id	21
title	“Legacy Set”
description	None
type	BONUS
color	BLUE
name	“?”
x_position	250
y_position	2000
leetcode_url	None
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Distinct Subsequences” - “Regular Expression Matching” - “Wildcard Matching”
requiredBy	- “Best Time to Buy and Sell Stock (Sliding Window)” - “Is Subsequence (Two Pointers)” - “Generate Parentheses (Backtracking)” - “Longest Valid Parentheses (Stacks)”

	- “Binary Tree Maximum Path Sum (Trees)” - “Trapping Rain Water (Two Pointers)”
flip_arrow	TBD

id	22
title	“Best Time to Buy and Sell Stock (Sliding Window)”
description	“You are given an array prices where prices[i] is the price of a given stock on the ith day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock. Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0.”
type	BONUS
color	BLUE
name	“!”
x_position	150
y_position	2250
leetcode_url	“https://leetcode.com/problems/best-time-to-buy-and-sell-stock/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD

id	23
title	“Is Subsequence (Two Pointers)”
description	“Given two strings s and t, return

	true if s is a subsequence of t, or false otherwise.”
type	BONUS
color	BLUE
name	“@”
x_position	350
y_position	2250
leetcode_url	“https://leetcode.com/problems/is-subsequence/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD

id	24
title	“Generate Parentheses (Backtracking)”
description	“Given n pairs of parentheses, write a function to generate all combinations of well-formed parentheses.“
type	BONUS
color	BLUE
name	“#”
x_position	50
y_position	2500
leetcode_url	“https://leetcode.com/problems/generate-parentheses/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD

id	25
title	“Longest Valid Parentheses (Stacks)”
description	“Given a string containing just the characters '(' and ')', return the length of the longest valid (well-formed) parentheses substring.”
type	BONUS
color	BLUE
name	“$”
x_position	450
y_position	2500
leetcode_url	“https://leetcode.com/problems/longest-valid-parentheses/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD

id	26
title	“Binary Tree Maximum Path Sum”
description	“Given the root of a binary tree, return the maximum path sum of any non-empty path.”
type	BONUS
color	BLUE
name	“%”
x_position	150
y_position	2750
leetcode_url	“https://leetcode.com/problems/binary-tree-maximum-path-sum/description/”
world	“World 11 - Dynamic Programming”
world_id	11

prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD

id	27
title	“Trapping Rain Water (Two Pointers)”
description	“Given n non-negative integers representing an elevation map where the width of each bar is 1, compute how much water it can trap after raining.”
type	BONUS
color	BLUE
name	“^”
x_position	350
y_position	2750
leetcode_url	“https://leetcode.com/problems/trapping-rain-water/description/”
world	“World 11 - Dynamic Programming”
world_id	11
prerequisites	- “Legacy Set”
requiredBy	None
flip_arrow	TBD