**TECHNICAL SKILLS**

**TEXT BOOKS:-**

1. Brian W. Kernighan, Dennis M. Ritchie, “The C Programming Language: ANSI C Version”, 2/e,

Prentice-Hall/Pearson Education-2005.

1. E. Balagurusamy, **“**Programming in ANSI C” 4th ed., Tata McGraw-Hill Education, 2008.
2. R. F. Gilberg, B. A. Forouzan, “Data Structures”, 2nd Edition, Thomson India Edition-2005.

**REFERENCE BOOKS:-**

1. Mark Allen weiss, Data Structures and Algorithm Analysis in C, 2008, Third Edition, Pearson Education.
2. Horowitz, Sahni, Anderson Freed, “Fundamentals of Data structures in C”, 2nd Edition-2007.
3. Robert Kruse, C. L. Tondo, Bruce Leung, Shashi Mogalla, “Data structures and Program Design in C”,

4th Edition-2007.

1. C for Engineers and Scientists – An Interpretive Approach by Harry H. Cheng, Mc Graw Hill International Edition-2010.
2. Jeri R. Hanly, Elliot B. Koffman, “Problem Solving and Program Design in C”, 7/e, Pearson Education-2004.
3. Jean Paul Trembly Paul G.Sorenson, “An Introduction to Data Structures with applications”, 2nd Edition.

**OTHER BOOKS, REFERENCES: (As recommended for reference by the course team, if any): Nill**

**WEB REFERNCES/MOOCs:**

1. [**www.hackerrank.com**](http://www.hackerrank.com/)
2. [**www.spoj.com**](http://www.spoj.com/)
3. [**www.geeksforgeeks.com**](http://www.geeksforgeeks.com/)
4. [**www.hackerearth.com**](http://www.hackerearth.com/)
5. [**www.cprogramiz.com**](http://www.cprogramiz.com/)
6. [**www.codechef.com**](http://www.codechef.com/)

**Session**

**No.**

1

2

**Topic**

Design an algorithm and draw flowchart to print Student ID and Name

Design an algorithm and draw flowchart to compute area of circle considering radius as input.

Design an algorithm and draw flowchart to convert given Celsius temperature to Fahrenheit.

Design an algorithm and draw flowchart to check if a given number is even or odd.

Design an algorithm and draw flowchart to find the smallest among 3 given numbers.

Given the marks obtained by the students, maximum marks and pass marks in three subject’s, design an algorithm and draw flowchart to find whether the student passed or

not, if the student passes determine the percentage marks and grade. The grade is determined as follows:

1. percentage marks >=80 grade is A
2. percentage marks >=70 and <80 grade is

B

1. percentage marks >=60 and <70 grade is

C

1. percentage marks >=50 and <60 grade is

D

1. percentage marks <50 grade is F

Design and algorithm to calculate grade of the student by reading three subject marks.

|  |  |  |  |
| --- | --- | --- | --- |
| **BTL** |  | **Teaching-** |  |
| **Reference** | **Learning** |  |
| **Level** |  |
|  | **Methods** |  |
|  |  |  |
| 1 | W REF[5] |  |  |
| 1 | W REF[5] | Discussion/ |  |
|  |  |  |
|  |  | Practice |  |
| 1 | W REF[5] |  |  |

|  |  |
| --- | --- |
| 1 | W REF[5] |

1

1

W REF[5]

Discussion/

Practice

W REF[5]

3

4

5

Design algorithm and flowchart to pint first n natural numbers.

Design algorithm and flowchart to print multiplication table for a given number.

Design algorithm and draw flowchart to check whether a given number is a Perfect Number or not.

Hello World

"Hello World!" in C

Day 1: Data Types

Playing with Characters

Operators

Sum and Difference of 2 numbers

Operators

Find Product

Let Us Understand Computer

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | W REF[5] |  |  |
| 1 | W REF[5] | Discussion/ |  |
|  |  | Practice |  |
| 1 | W REF[5] |  |  |
|  |  |  |

|  |  |
| --- | --- |
| 1 | W REF[1]- Hello World |

|  |  |  |
| --- | --- | --- |
| 1 | W REF[1]- "Hello World!" in C | Discussion/ |
|  |  | Practice |

|  |  |  |
| --- | --- | --- |
| 2 | W REF[1]-Data Types |  |
|  |  |

1. W REF[1]- Playing with

Characters

|  |  |  |
| --- | --- | --- |
| 1 | W REF[1]- Operators |  |
|  |  |

1. W REF[1]- Sum and Difference

of 2 numbers

|  |  |  |
| --- | --- | --- |
| 2 | W REF[4]-Operators | Discussion/ |
|  |  | practice |

1. W REF[4]- Find Product

|  |  |  |
| --- | --- | --- |
| 2 | W REF[4]- Fin Let Us |  |
| Understand Computer |  |
|  |  |

Cutting paper squares

Maximum height Triangle

Birthday Party

6

Count Numbers

Count Divisors

Boxes through a Tunnel

Introduction to Conditional Statements

Conditional Statements in C

Library Fine

7

Nested Logic Conditional Statements

Conditional Statements-1

Conditional Statements-2

Loops

Binary Numbers

Maximum Draws

1. For loop in C Handshake Utopian Tree

|  |  |  |
| --- | --- | --- |
| 1 | W REF[1]- Cutting paper squares |  |
| 2 | W REF[1]- Maximum height |  |
| Triangle |  |

1. W REF[4]- Birthday Party

|  |  |  |  |
| --- | --- | --- | --- |
| 1 |  | Discussion/ |  |
| W REF[4]- Count Numbers | Practice |  |
|  |  |  |

1. W REF[4]- Count Divisors
2. W REF[1]- Boxes through a

Tunnel

1. W REF[1]- Introduction to

Conditional Statements

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | W REF[1]- |  |  |
|  |  |  |
|  | Conditional Statements in C |  |  |
| 2 | W REF[1]- Library Fine |  |  |
|  |  |  |
| 2 | W REF[1]- Nested Logic | Discussion/ |  |
| Conditional Statements |  |
|  | Practice |  |
|  |  |  |
| 1 | W REF[1]- Conditional |  |  |
| Statements-1 |  |  |
|  |  |  |
| 2 | W REF[1]- Conditional |  |  |
| Statements-2 |  |  |
|  |  |  |
|  |  |  |  |
| 2 | W REF[1]- Loops |  |  |
| 1 | W REF[1]- Binary Numbers |  |  |
| 1 | W REF[1]- Maximum Draws |  |  |
| 1 | W REF[1]- For loop in C |  |  |
|  | Discussion/ |  |
|  |  |  |
|  |  | Practice |  |
| 1 | W REF[1]- Handshake |  |  |
|  |  |  |
| 2 | W REF[1]- Utopian Tree |  |  |
|  |  |  |

Sum of Digits of a Five Digit Number

Angry Professor

Bitwise AND

Bitwise Operators

9

Flipping Bits

Pikachu vs Team Meowstic and Helping Hand

Find Digits

Running time & Complexity

Staircase

Printing Pattern using Loops

10

Digital Sequence

Fascinating Number

Day 7: Arrays

Simple Array Sum

Very Big Sum

11

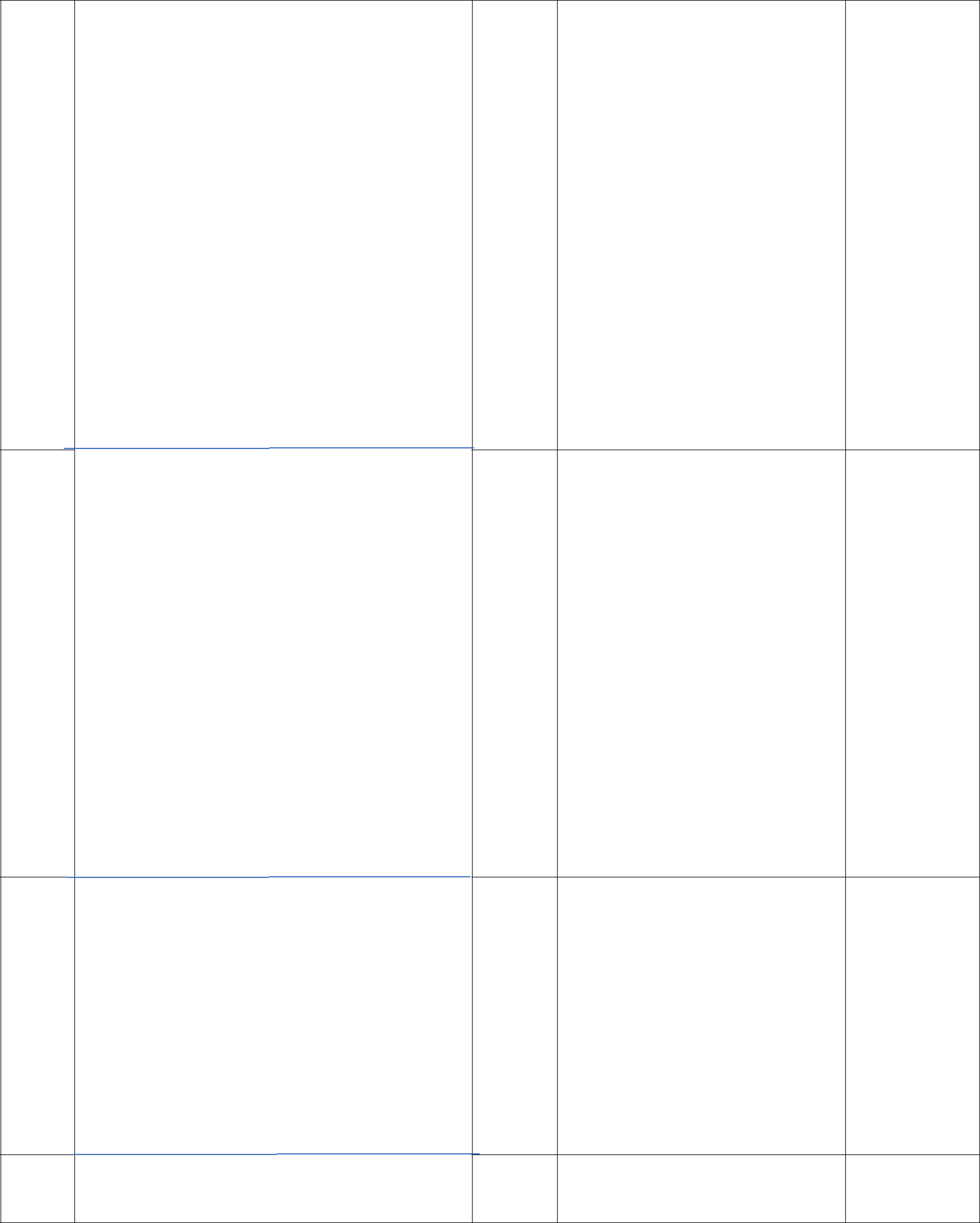
Arrays

Professor and his operations

Chef and Glove

1. W REF[1]- Sum of Digits of a

Five Digit Number



1. W REF[1]- Angry Professor
2. W REF[1]- Bitwise AND

|  |  |  |
| --- | --- | --- |
| 2 | W REF[1]- Bitwise Operators |  |
|  | Discussion/ |  |
|  | Practice |  |
| 3 | W REF[1]- Flipping Bits |  |
|  |  |

1. W REF[4]- Pikachu vs Team Meowstic and Helping Hand

|  |  |
| --- | --- |
| 2 | W REF[1]- Find Digits |

1. W REF[1]- Running time &

Complexity

|  |  |  |
| --- | --- | --- |
| 2 | W REF[1]- Staircase |  |
| 3 | W REF[1]- Printing Pattern using |  |
| Loops |  |
|  | Discussion/ |  |
|  | Practice |  |
| 2 | W REF[4]- Digital Sequence |  |
|  |  |
| 2 | W REF[3]- Fascinating Number |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | W REF[1]- Arrays |  |  |
| 1 | W REF[1]- Simple Array Sum |  |  |
| 1 | W REF[1]- Very Big Sum |  |  |
| 2 | W REF[3]- Very Big Sum | Discussion/ |  |
|  | Practice |  |
|  |  |  |

1. W REF[4]- Professor and his

operations

|  |  |  |
| --- | --- | --- |
| 2 | W REF[6]- Chef and Glove |  |
|  |  |

1. Grading Students Birthday Cake Candles Plus Minus

Missing Numbers

C Program to Generate Multiplication Table Minimum Loss

1D Arrays in C

Mini-Max Sum

Day 20: Sorting

13

Student Arrangement

Find an element in hidden array

Between Two Sets

Day 6: Let’s Review

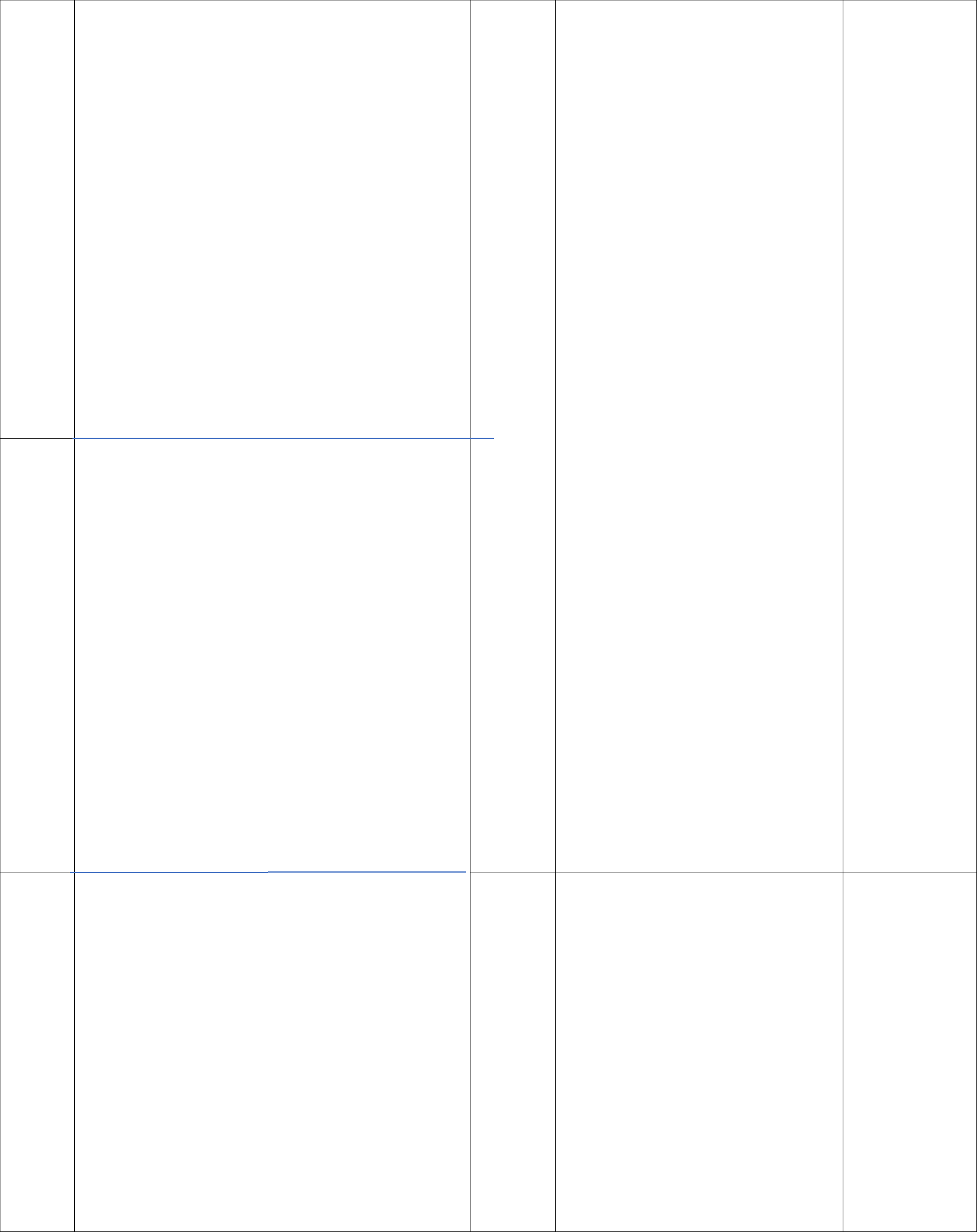
Time Conversion

14

Strong Password

Caesar Cipher

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | W REF[1]- Grading Students |  |  |
| 1 | W REF[1]- Birthday Cake | Discussion/ |  |
|  | Candles |  |
|  |  | Practice |  |
| 1 | W REF[1]- Plus Minus |  |  |
| 1 | W REF[3]- Missing Numbers |  |  |
| 2 | W REF[5]- C Program to |  |  |
| Generate Multiplication Table |  |  |
|  |  |  |
| 2 | W REF[1]- Minimum Loss |  |  |
|  |  |  |
|  |  |  |  |
| 1 | W REF[1]- 1D Arrays in C |  |  |



|  |  |  |  |
| --- | --- | --- | --- |
| 1 | W REF[1]- Mini-Max Sum |  |  |
|  |  |  |
| 2 | W REF[1]- Day 20: Sorting |  |  |
|  | Discussion/ |  |
|  |  |  |
|  | W REF[4]- Student Arrangement | Practice |  |
| 2 |  |  |

1. W REF[3]- Find an element in

hidden array

1. W REF[1]-Between Two Sets
2. W REF[1] – Let’s Review
3. W REF[1] – Time Conversion

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Discussion/ |  |
| 1 | W REF[1] – Strong Password | Practice |  |
| 2 | W REF[1] – Caesar Cipher |  |  |
|  |  |  |

VowelPhobia

Printalternate elements of an array

Printing Tokens

Digit Frequency

Diagonal Difference

Day 11: 2D Arrays

15

Batman and Tick-tack-toe

Counting Valleys

Lonely Integer

Maximizing XOR

Array Reversal

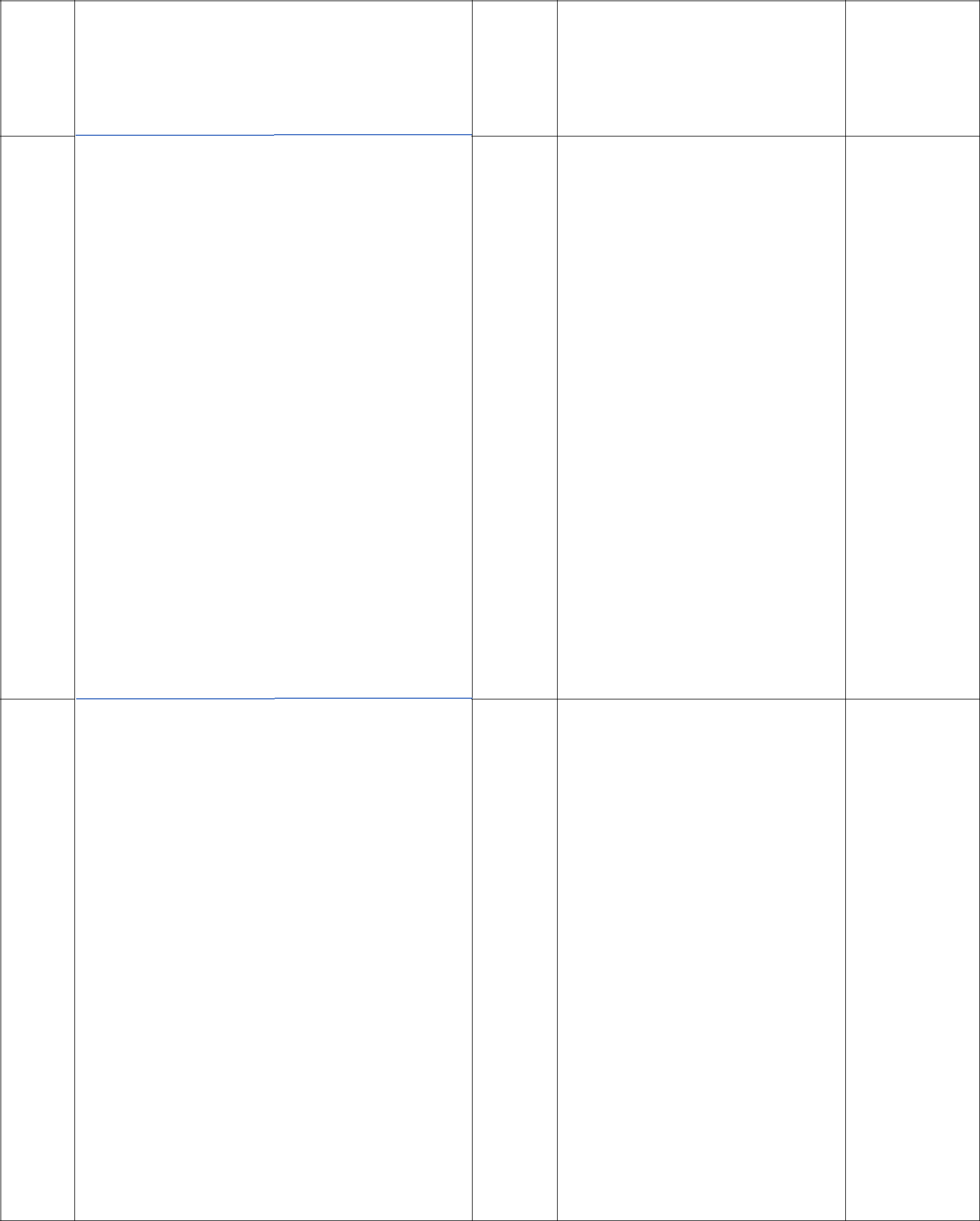
16

Year 3017

Hackers with Bits

Short Name

|  |  |
| --- | --- |
| 3 | W REF[4] – VowelPhobia |



1. W REF[3] - Print alternate

elements of an array

1. W REF[1] – Printing Tokens
2. W REF[1] – Digit Frequency

1 W REF[1] – Diagonal Difference

|  |  |  |
| --- | --- | --- |
| 2 | W REF[1] – Day 11: 2D Arrays | Discussion/ |
|  |  | Practice |

1. W REF[1] – Batman and Tick-

tack-toe

2 W REF[1] – Counting Valleys

1. W REF[1] – Lonely Integer
2. W REF[1] – Maximizing XOR

|  |  |  |
| --- | --- | --- |
| 1 | W REF[1] – Array Reversal |  |
|  | Discussion/ |  |
|  | Practice |  |
| 3 | W REF[6] – Year 3017 |  |
| 2 | W REF[4] – Hackers with Bits |  |
|  |  |
| 2 | W REF[4] – Short Name |  |
|  |  |

17

18

19

Arrays-2

Arrays-3

Array-4

Array-5(Manchurian And Trivisible Arrays)

Xsquare And Two Strings

ARRAYSUB - subarrays

Compare the Triplets

Constructing a number

Palindrome

Toggle String

Recursive Function

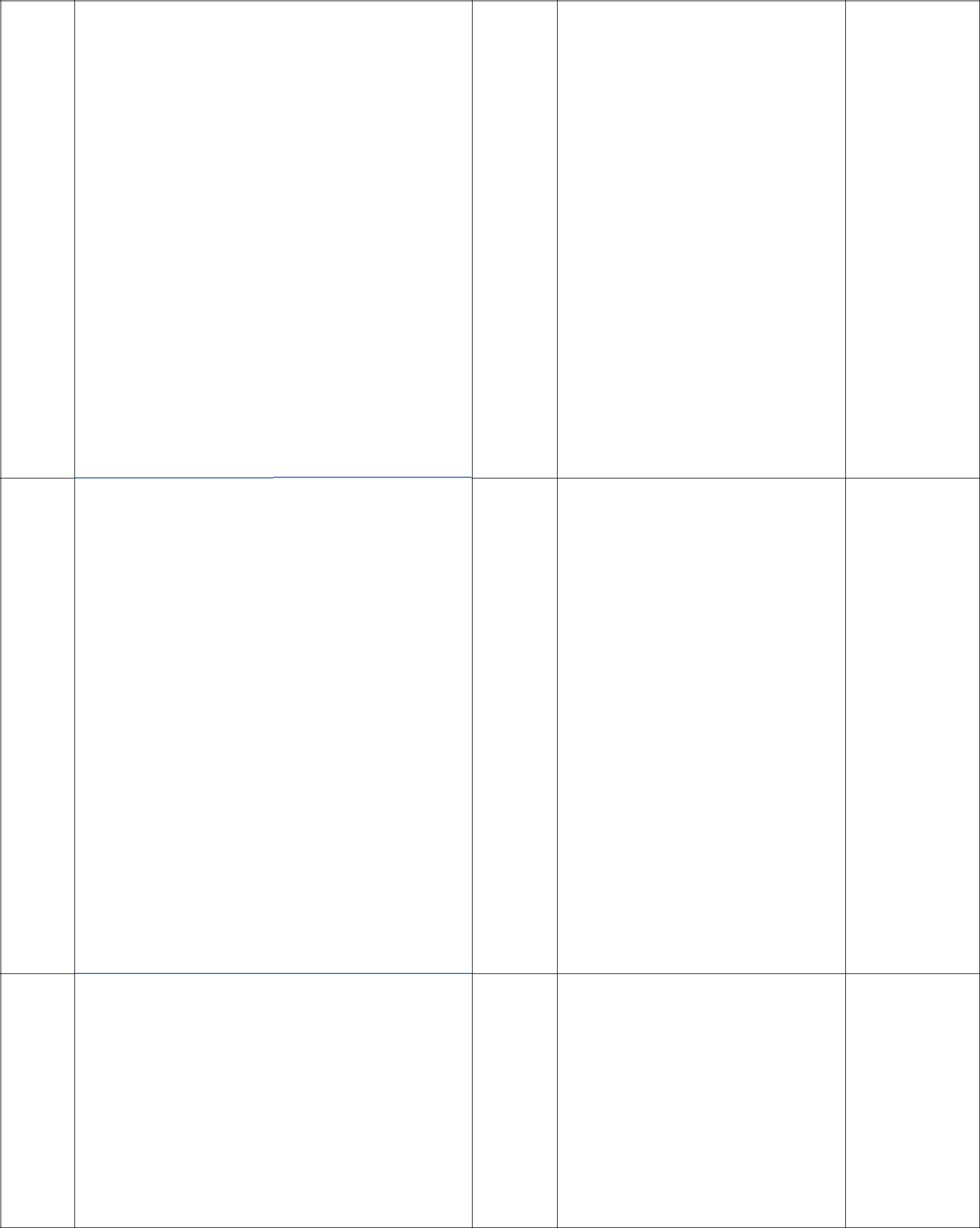
Binary vs Linear Search

Find the point

Calculate the Nth Term

Strike or Spare

|  |  |  |
| --- | --- | --- |
| 1 | W REF[1] – Arrays - 2 |  |
| 2 | W REF[1] – Arrays - 3 |  |
| 2 | W REF[6] – Array - 4 |  |
| 1 | W REF[4] – Array - 5 |  |
|  |  |



1. W REF[4] - Xsquare And Two

Strings

1. W REF[2] - ARRAYSUB -

subarrays

1. W REF[1] - Compare the Triplets
2. W REF[2] - Constructing a

number

|  |  |
| --- | --- |
| 2 | W REF[4] - Palindrome |

|  |  |  |
| --- | --- | --- |
| 1 | W REF[4]- Toggle String | Discussion/ |
|  |  | Practice |
| 3 | W REF[4] - Recursive Function |  |

1. W REF[4] - Binary vs Linear

Search

1. W REF[1] - Find the point
2. W REF[1] - Calculate the Nth

Term

|  |  |  |
| --- | --- | --- |
| 2 | W REF[6] – Strike or Spare |  |
|  |  |

20

21

Signal Range

MUL-Fast Multiplication

Monk and Order of Phoenix

Recursion

Simran and stairs

Policemen and thieves

Speed

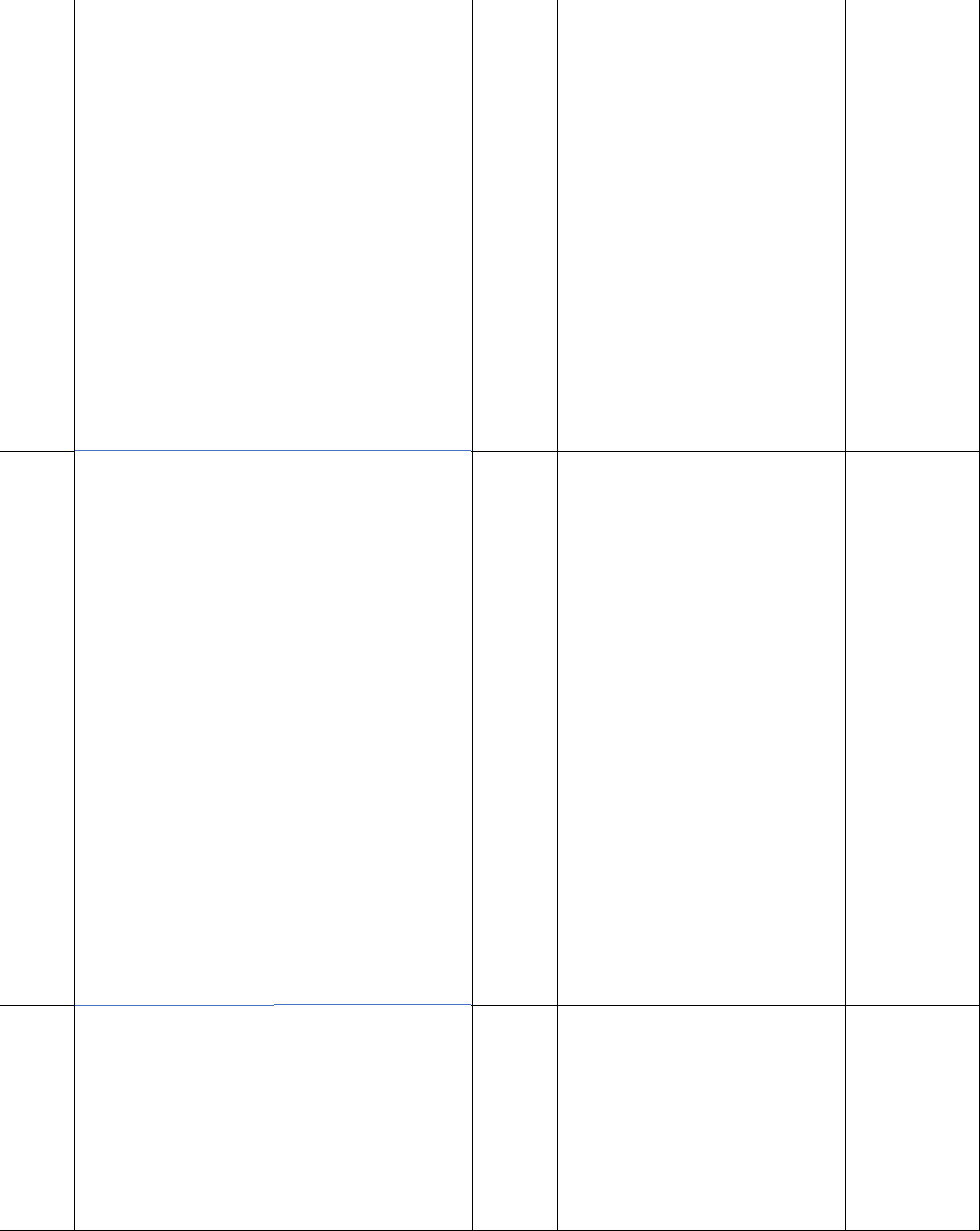
Mark The Answer

Weighing the Stones

Small Triangles, Large Triangles

GCD Strings

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | W REF [4] – Signal Range |  |  |
|  | W REF[2] - MUL-Fast | Discussion/ |  |
| 3 | Multiplication | Practice |  |
| 3 | W REF[4] - Monk and Order of |  |  |
| Phoenix |  |  |
|  |  |  |



|  |  |  |  |
| --- | --- | --- | --- |
| 3 | W Ref[1] - Recursion |  |  |
| 2 | W Ref[4] - Simran and stairs |  |  |
| 2 | W Ref [4] - Policemen and |  |  |
| thieves |  |  |
|  | Discussion/ |  |
|  |  |  |
|  |  | Practice |  |
| 2 | W Ref [4] - Speed |  |  |
|  |  |  |

1. W Ref [4] - Mark The Answer
2. W Ref[4] - Weighing the Stones
3. W Ref[1] - Small Triangles, Large

Triangles

1. W Ref[4] - GCD Strings

22

23

Roy and Profile Picture

Two Strings

ArrayGame

Post Transition

Roy and Profile Picture

Minimize Cost

Round Table Killers

City group

Easy Multiple

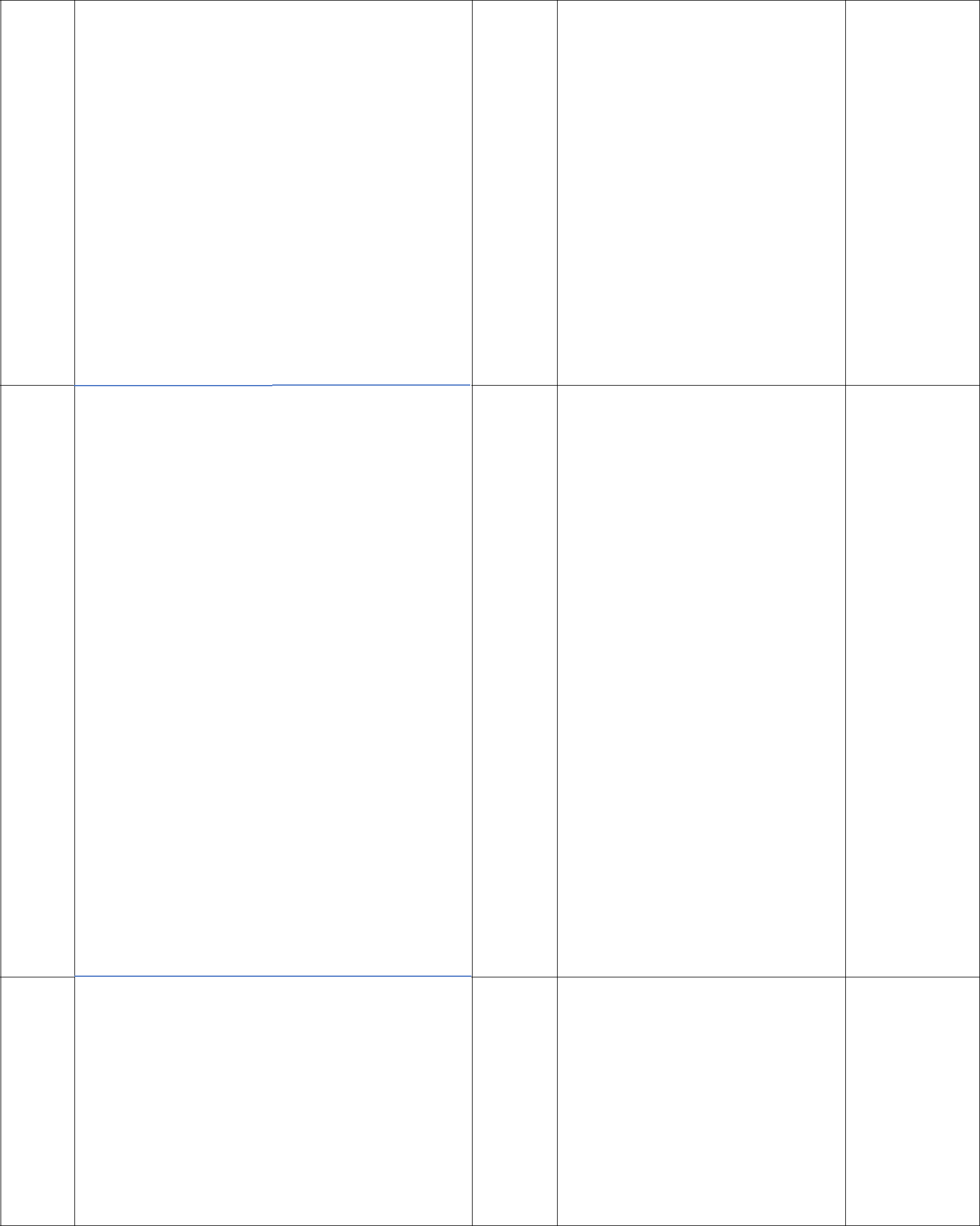
Structuring the document

Recursive Sums

Counting Strings

1. W Ref[4] - Roy and Profile

Picture



|  |  |  |
| --- | --- | --- |
| 2 | W Ref[4] - Two Strings | Discussion/ |
|  |  | Practice |

|  |  |
| --- | --- |
| 3 | W Ref[4] - ArrayGame |

|  |  |
| --- | --- |
| 3 | W Ref [1] - Post Transition |

1. W Ref [4] - Roy and Profile Picture

|  |  |  |
| --- | --- | --- |
| 1 | W Ref [4] - Minimize Cost |  |
|  | Discussion/ |  |
|  | Practice |  |
| 1 | W Ref [4] - Round Table Killers |  |
|  |  |
| 2 | W Ref [4] - City group |  |
|  |  |
| 1 | W Ref [4] - Easy Multiple |  |
|  |  |

1. W Ref [1] - Structuring the

document

|  |  |  |
| --- | --- | --- |
| 1 | W Ref [4] - Recursive Sums | Discussion/ |
|  |  | Practice |

|  |  |
| --- | --- |
| 3 | W Ref [4] - Counting Strings |

24

25

Play with numbers

Life, the Universe, and Everything

Seating Arrangement

Maximum Element

Help your roommate

Subset AND

Lucky Numbers

Poisonous gas

Game of Deletion

Queue using two stacks

Download file

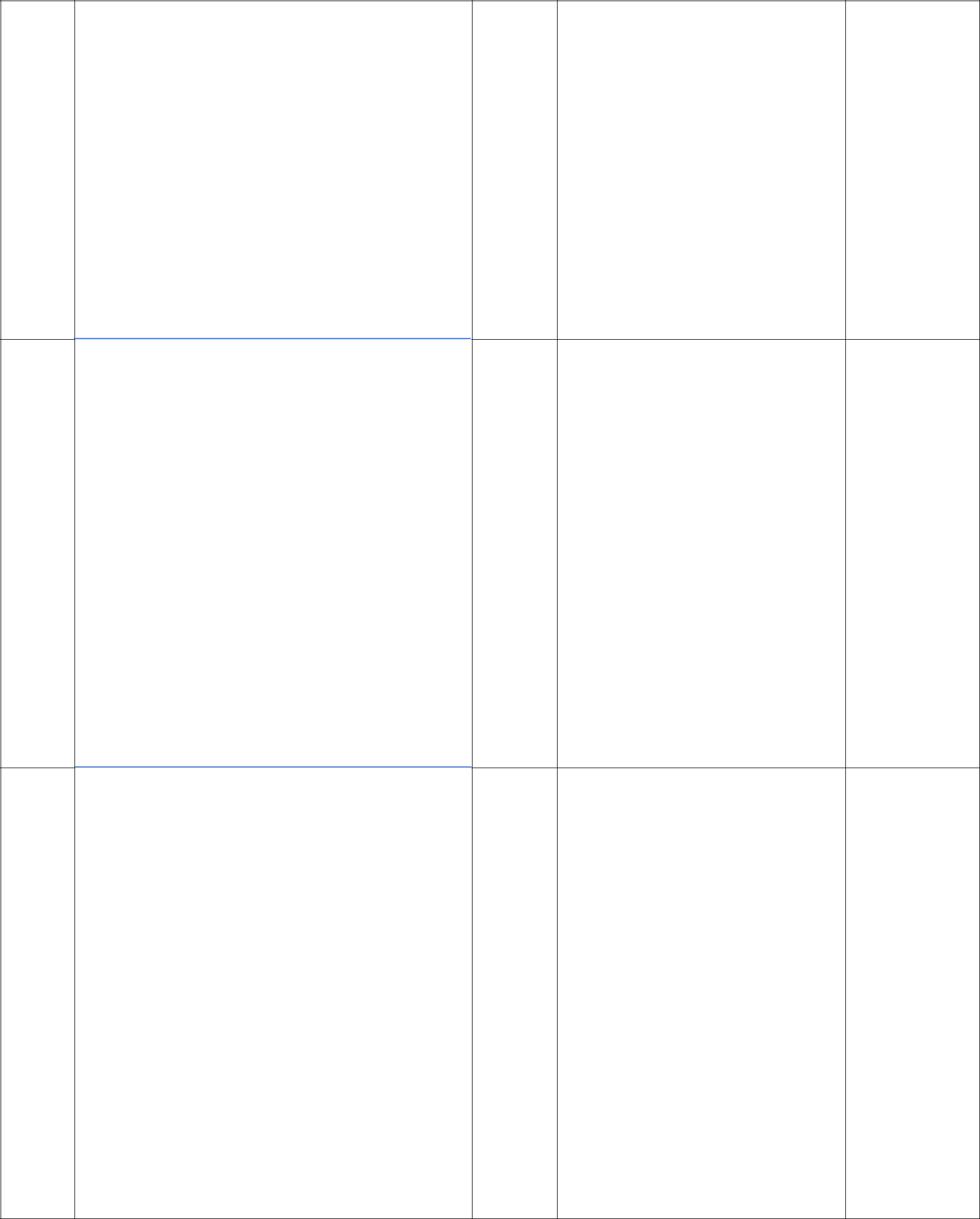
Small Factorials

Coins

Discover the Monk

Maximum Sum

1 W Ref [4] - Play with numbers



|  |  |  |
| --- | --- | --- |
| 2 | W Ref [4] - Life, the Universe, | Discussion/ |
|  | and Everything | Practice |

3 W Ref [4] - Seating Arrangement

1. W Ref [1] - Maximum Element
2. W Ref [4] - Help your roommate

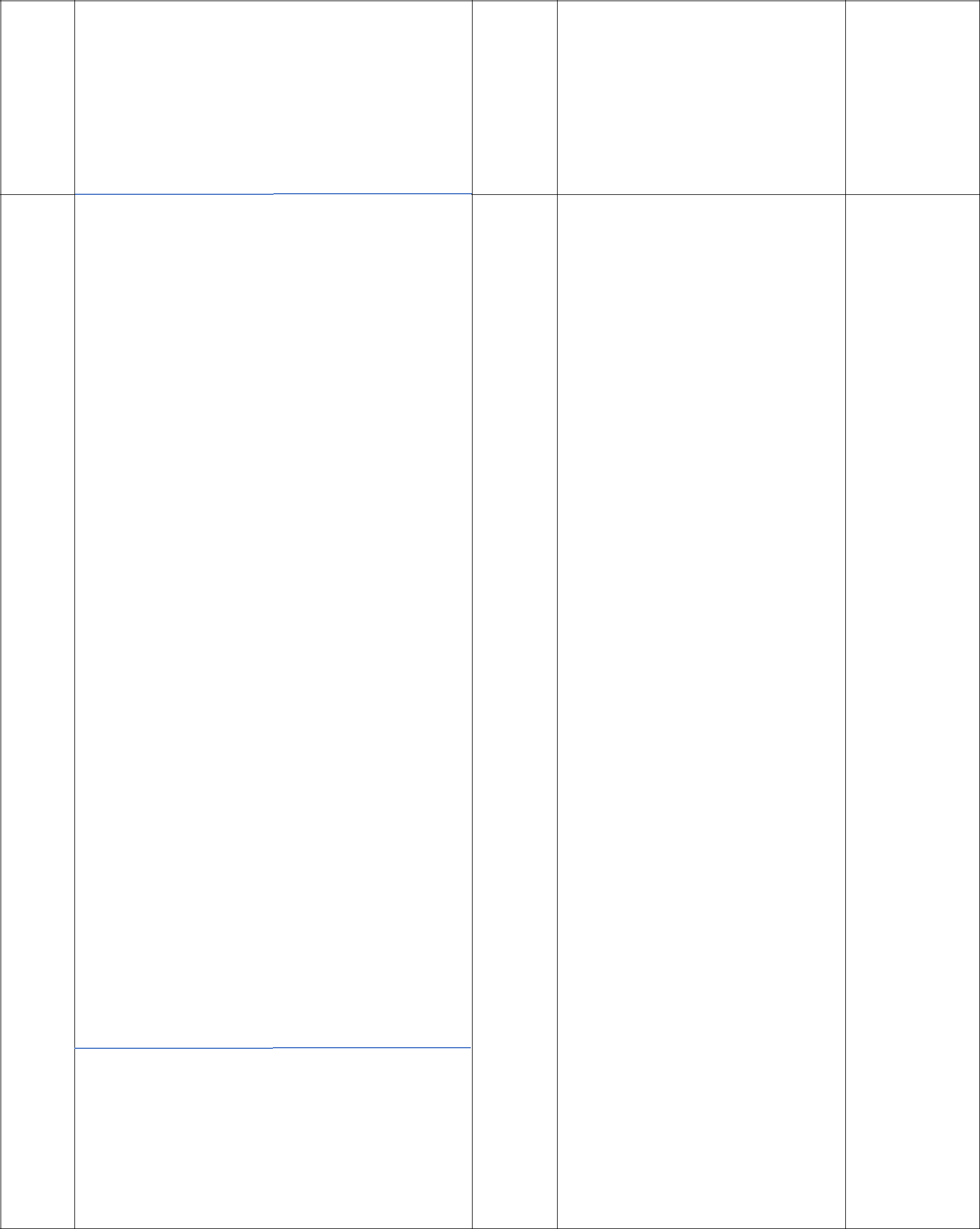
|  |  |
| --- | --- |
| 2 | W Ref [4] – Subset AND |
|  | Discussion/ |
|  | Practice |
| 2 | W Ref [4] – Lucky Numbers |

1. W Ref [4] – Poisonous gas
2. W Ref [4] – Game of Deletion
3. W Ref [1] - Queue using two

stacks

1. W Ref [4] - Download file

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2 | W Ref [4] – Small Factorials | |  |  |
|  |  |  |
|  |  |  | Discussion/ |  |
| 2 | W Ref [4] – Coins | | Practice |  |
|  |  |  |
| 2 | W Ref [4] – | Discover the Monk |  |  |
|  |  |  |
| 2 | W Ref [4] – Maximum Sum | |  |  |
|  |  |  |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 26 | Insert a node at a specific position in linked | 3 | W Ref [1] - Insert a node at a | Discussion/ |  |
| list | specific position in linked list |  |
|  |  | Practice |  |
|  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 27 | | Print in reverse | 3 | W Ref [1] - | Print in reverse | Discussion/ | |  |  |
|  |  |  |  |  |  |  | Practice | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | 28 | | Get node value | 3 | W Ref [1] - | Get node value | Discussion/ | |  |  |
|  |  |  |  |  |  |  | Practice | |  |  |
|  | 29 | | Compare two linked lists | 3 | W Ref [1] - Compare two | | Discussion/ | |  |  |
|  | linked lists | |  |  |
|  |  |  |  |  |  |  | Practice | |  |  |
|  |  |  |  |  |  | |  |  |  |  |
|  | 30 | | Merging Two Linked Lists | 3 | W Ref [3] - Merging Two | | Discussion/ | |  |  |
|  | Linked Lists | |  |  |
|  |  |  |  |  | Practice | |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | |  |  |  |  |
|  |  |  | Queue using two stacks | 3 | W Ref [1] - Queue using two | |  |  |  |  |
|  |  |  | stacks | |  |  |  |  |
|  | 31 | |  | Discussion/ | |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  | 3 | W Ref [4] - Chocolate | | Practice | |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | Chocolate Distribution | Distribution | |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | |  |  |  |  |
|  |  |  | Monk and Chamber of Secrets | 3 | W Ref [4] - Monk and | |  |  |  |  |
|  |  |  |  | Chamber of Secrets | |  |  |  |  |
|  | 32 | |  |  |  |  |  |  |  |  |
|  |  |  |  |  | W Ref [4] - Hacker and traffic | | Discussion/ | |  |  |
|  |  |  | Hacker and traffic lights | 3 | Practice | |  |  |
|  |  |  |  | lights |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | |  |  |  |  |
|  |  |  | Frustrated coders | 2 | W Ref [4] - Frustrated coders | |  |  |  |  |
|  | 33 | | Stack operations | W Ref [4] - Stack operations | |  |  |  |  |
|  | 2 | Discussion/ | |  |  |
|  |  |  | A Game of Numbers | W Ref [4] – Game of Numbers | |  |  |
|  |  |  | 2 |  |  |
|  |  |  | Monk and Prisoner of Azkaban | W Ref [4] – Monk and Prisoner | | Practice | |  |  |
|  |  |  | 2 |  |  |
|  |  |  |  |  |  |  |
|  |  |  | Insert a node at a specific position in linked |  |  |  |  |  |  |  |
|  |  |  | 3 | W Ref [1] – Insert a node | |  |  |  |  |
|  |  |  | list |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | 34 | |  |  |  |  | Discussion/ | |  |  |
|  |  |  | Print in reverse | 3 | W Ref [1] – Print in reverse | | Practice | |  |  |
|  |  |  | Delete a Node | 1 | W Ref [1] – Delete a Node | |  |  |  |  |
|  |  |  |  |  |  |  |  |

Get node value

Merge two sorted linked lists

35

Delete duplicate-value nodes from a sorted linked list

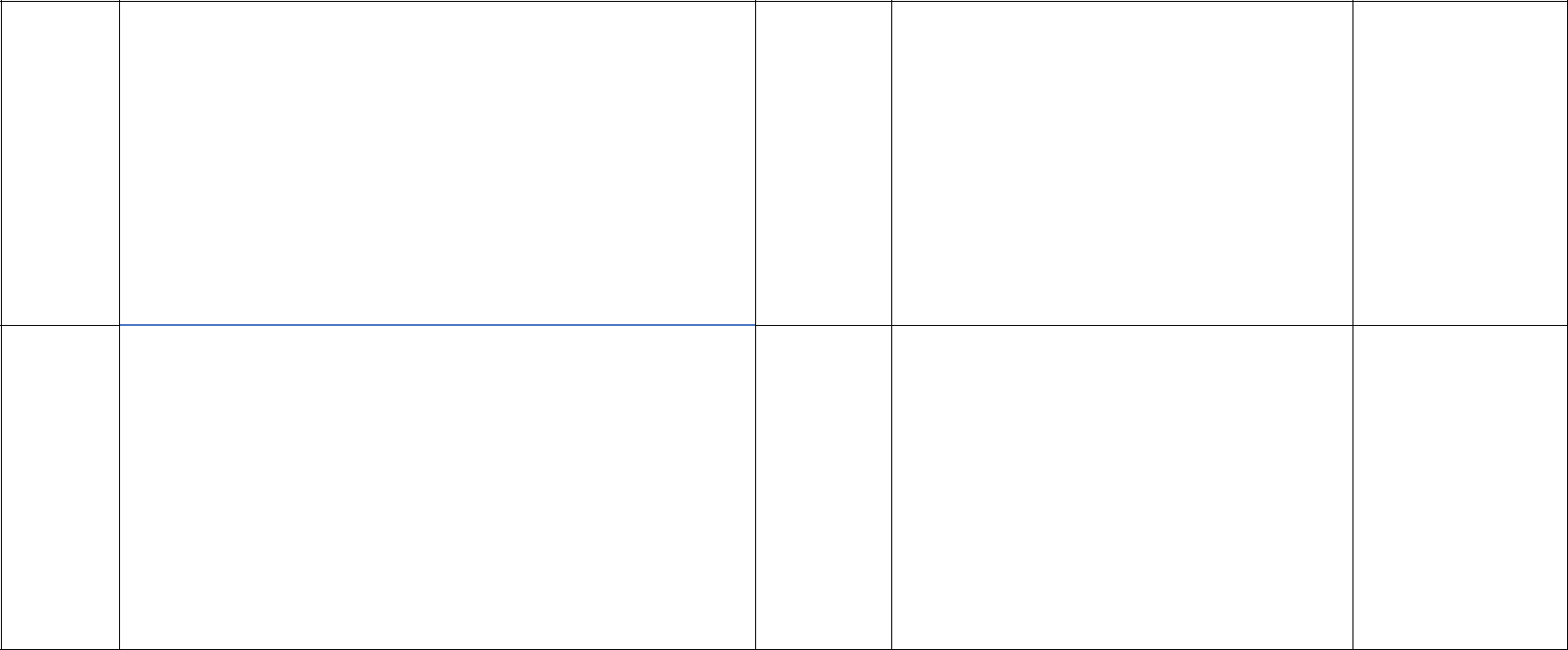
Compare two linked lists

Rotate a Linked List

36

Delete N nodes after M nodes of a LL

1. W Ref [1] – Get node value



|  |  |  |  |
| --- | --- | --- | --- |
| 2 | W Ref [1] – Merge two sorted |  |  |
|  | Discussion/ |  |
|  | lists |  |
|  | Practice |  |
|  | W Ref [1] – Delete Duplicate |  |
| 1 |  |  |

1. W Ref [1] – Compare two linked

lists

|  |  |  |  |
| --- | --- | --- | --- |
| 3 | W Ref [3] – Rotate a Linked List | Discussion/ |  |
|  |  |  |
| 3 | W Ref[3] – Delete N nodes of a | Practice |  |
|  | Linked List |  |  |

**GENERAL INSTRUCTIONS**

Students should come prepared for classes and carry the text book(s) or material(s) as prescribed by the Course Faculty to the class.

**NOTICES**

Most of the notices are available on the LMS platform.

All notices will be communicated through the institution email.

All notices concerning the course will be displayed on the respective Notice Boards.