First Part: Basic Questions

1. Print a right-angled triangle of numbers.
2. Print an inverted right-angled triangle of numbers.
3. Print a pyramid of numbers.
4. Print an inverted pyramid of numbers.
5. Print Floyd's triangle.
6. Print numbers in a square grid.
7. Print numbers in increasing order.
8. Print numbers in decreasing order.
9. Print a right-angled triangle of stars.
10. Print an inverted right-angled triangle of stars.
11. Print a mirrored right-angled triangle.
12. Print a pyramid of stars.
13. Print an inverted pyramid of stars.
14. Print a hollow square of stars.
15. Print a hollow right-angled triangle.
16. Print natural numbers from 1 to 10.
17. Print natural numbers from 10 to 1.
18. Print natural numbers from 1 to 10 except 5.
19. Print natural numbers from 1 to 10 except 3 and 8.
20. Print natural numbers from 1 to 10 except 5 using continue.
21. Print natural numbers from 1 to 10 except 5 using continue in a while loop.
22. Print natural numbers from X to Y (user-defined X and Y).
23. Print even numbers from X to Y.
24. Print odd numbers from X to Y.
25. Print numbers from X to Y divisible by 5.
26. Print the multiplication table of a number.
27. Print the multiplication table of 8.
28. Print alphabets from A to Z.
29. Print ASCII values of alphabets from A to Z.
30. Sum of natural numbers from 1 to 10.

31. Sum of numbers from X to Y (user-defined).
32. Product of natural numbers from 1 to 10.
33. Product of numbers from X to Y (user-defined).
34. Check if a number is positive or negative.
35. Check if a number is even or odd.
36. Check if a number is divisible by another number.
37. Add two numbers.
38. Subtract two numbers.
39. Multiply two numbers.
40. Divide two numbers.
41. Find the remainder of two numbers.
42. Swap two values using a third variable.
43. Swap two values without using a third variable.
44. Print the factorial of a given number.
45. Print the exponential value for a given base and power.
46. Calculate the area of a circle given its radius.
47. Calculate the circumference of a circle given its radius.
48. Calculate the area of a rectangle given its sides.
49. Calculate the area of a triangle given its sides.
50. Calculate the area of a square given its side length.
51. Convert Celsius to Fahrenheit.
52. Convert Fahrenheit to Celsius.
53. Print all elements in an array.
54. Print all even elements in an array.
55. Print all odd elements in an array.
56. Print all elements at even indices in an array.
57. Print all odd elements at even indices in an array.
58. Print the elements of an array in reverse order.
59. Calculate the sum of all elements in an array.
60. Calculate the sum of all even elements in an array.
61. Calculate the sum of all odd elements in an array.

62. Calculate the sum of the first and last elements in an array.
63. Calculate the sum of the last two elements in an array.
64. Calculate the multiplication of all elements in an array.
65. Calculate the average value of all elements in a given array.
66. Print the largest element in a given array.
67. Print the smallest element in a given array.
68. Print the second largest element in an array.
69. Print the second smallest element in an array.
70. Reverse a string without using built-in methods.
71. Reverse each word in a string individually.
72. Reverse a string while maintaining the position of spaces.
73. Check if a string is a palindrome.
74. Count the number of vowels and consonants in a string.
75. Count the frequency of each character in a string.
76. Remove duplicate characters from a string.
77. Replace duplicate characters with "".
78. Convert the first character of each word in a string to uppercase.
79. Count the number of words in a string.
80. Replace all spaces in a string with '%20'.
81. Remove a specific character from a string.
82. Check if two strings are anagrams of each other.
83. Check if a string contains only digits.
84. Convert a string to a character array.
85. Convert a string to its ASCII representation.
86. Implement your own toLowerCase() and toUpperCase() methods.
87. Replace all vowels in a string with a given character.
88. Check if a given number is prime
89. Print factors of a given number.
90. Sum of factors of a given number.
91. Check if a number is a SPY number.

Second Part: Advanced Questions

1. Print a diamond of numbers.
2. Print a hollow diamond of numbers.
3. Print concentric squares with numbers.
4. Print a sandglass pattern of numbers.
5. Print alternating rows of numbers.
6. Print a checkerboard pattern of numbers.
7. Print a Z-shape of numbers.
8. Print a diamond pattern.
9. Print a hollow diamond pattern.
10. Print a half-diamond pattern.
11. Print a zig-zag star pattern.
12. Print a plus (+) shape using stars.
13. Print an hourglass pattern of stars.
14. Print username 25 times (using the Scanner Class).
15. Print username a user-defined number of times.
16. Sum of even factors of a given number.
17. Sum of odd factors of a given number.
18. Count the factors of a given number.
19. Print prime numbers between 1 to 100.
20. Print the first n prime numbers (e.g., 50).
21. Print the nth prime number.
22. Print the next prime number for a given number.
23. Check if a given number is a perfect number.
24. Print perfect numbers between 1 to 50,000.
25. Print palindromes between 1 to 10,000 with serial numbers.
26. Print the first n palindromes (e.g., 50).
27. Print the nth palindrome.

28. Print the next palindrome number for a given number.
29. Print the first n SPY numbers.
30. Print the nth SPY number.
31. Print the next SPY number for a given number.
32. Check if a given number is a strong number.
33. Check if a given number is an Armstrong number.
34. Print the first n terms of the Fibonacci series.
35. Print the nth Fibonacci number.
36. Check if a number is a Fibonacci number.
37. Find the GCD or HCF of two numbers.
38. Find the LCM of two numbers.
39. Find the GCD or HCF of three numbers.
40. Find the LCM of three numbers.
41. Count the number of digits in a given number.
42. Print the sum of all the digits in a given number.
43. Count and print the even and odd digits separately.
44. Reverse a given number.
45. Find the largest digit in a given number.
46. Find the smallest digit in a given number.
47. Find the sum of factorials of digits in a given number.
48. Print the sum of a geometric progression.
49. Print the sum of an arithmetic progression.
50. Print the nth term of an arithmetic sequence.
51. Print the nth term of a geometric sequence.
52. Print the Lucas series up to n terms.
53. Generate the first n terms of a Tribonacci sequence.
54. Print the factorial series up to n terms.
55. Generate the Collatz sequence for a number.
56. Find the roots of a quadratic equation.
57. Find the binary representation of a number.
58. Find the decimal representation of a binary number.

59. Perform bitwise AND of two numbers.
60. Perform bitwise OR of two numbers.
61. Perform bitwise XOR of two numbers.
62. Check if a number is a power of 2.
63. Count the number of set bits in a number.
64. Count how many even elements are present in an array.
65. Calculate the sum of all prime numbers in a given array.
66. Calculate the multiplication of all even elements in an array.
67. Calculate the multiplication of all prime numbers in a given array.
68. Print all duplicate elements in an array.
69. Print duplicate names in a given array.
70. Print even duplicate elements in an array.
71. Print duplicate elements at even indices in an array.
72. Merge two arrays and find duplicate elements in the merged array.
73. Calculate the sum of duplicate elements in a given array.
74. Remove duplicate elements from an array.
75. Remove the most repeated elements from an array.
76. Remove the most repeated even elements from an array.
77. Remove duplicate elements from even indices in an array.
78. Sort an array without using predefined methods (Bubble Sort logic).
79. Sort an array using predefined methods (e.g., Arrays.sort()).
80. Sort an array in descending order using Bubble Sort logic.
81. Sort only the positive elements in a given array.
82. Sort only the even elements in a given array.
83. Sort only the prime elements in a given array.
84. Sort half the elements in a given array.
85. Merge two arrays and sort the merged elements.
86. Sort an array without using a third variable or predefined methods.
87. Sort only the first two and last two elements in a given array.
88. Calculate the sum of the largest and smallest elements in an array.
89. Calculate the multiplication of the largest and smallest elements in an array.

90. Calculate the average of the largest and smallest elements in an array.
91. Calculate the maximum difference between the largest and smallest elements in an array.
92. Calculate the minimum difference between the largest and smallest elements in an array.
93. Calculate the maximum difference between the second largest and second smallest elements in an array.
94. Print the nth largest element in a given array.
95. Print the nth smallest element in a given array.
96. Print the largest even element in a given array.
97. Print the smallest odd element in a given array.
98. Print the largest prime number in a given array.
99. Merge two arrays into a single array.
100. Merge only the even elements from two arrays into a single array and print it in reverse order.
101. Sort two arrays in ascending order, merge them, and print only the even numbers.
102. Sort the prime numbers from two arrays and merge them into a single array.
103. Merge the largest and smallest elements from two arrays into a single array.
104. Print all prime numbers in an array.
105. Calculate the sum of all prime numbers in a given array.
106. Calculate the sum of the last four even numbers in an array.
107. Count the number of elements in an array without using the .length property.
108. Replace each element with the greatest element on its right side.
109. Find the missing element in a given array.
110. Rearrange an array with alternate high and low elements.
111. Check if the sum of all array elements is an Armstrong number.
112. Check if the product of all array elements is a palindrome.
113. Check if the sum of all even elements is a strong number.
114. Check if the sum of all odd elements is an Armstrong number.
115. Reverse a given array using the swapping technique.
116. Shuffle a given array of integers.
117. Find the length of the longest consecutive element sequence in an unsorted array.
118. Write a Java Program for MergeSort.
119. Write a Java Program for QuickSort.

120. Convert all ArrayList elements into an array.
121. Convert all array elements into an ArrayList.
122. Replace every element with the next greatest element (from the right side) in a given array, replacing the last element with -1.
123. Reverse a string using recursion.
124. Find the longest palindrome in a string.
125. Find the longest palindromic substring in a string.
126. Generate all possible substrings of a string that are palindromes.
127. Check if a string is a rotation of a palindrome.
128. Count the number of palindromic substrings in a string.
129. Find all palindromic partitions of a string.
130. Find the first non-repeated character in a string.
131. Find the second most frequent character in a string.
132. Find the first repeating character in a string.
133. Check if a string contains duplicate adjacent characters.
134. Remove all characters that appear more than once in a string.
135. Find all substrings of a string.
136. Replace all occurrences of a substring in a string.
137. Find the index of the first occurrence of a substring in a string.
138. Find all repeated substrings in a string.
139. Find the longest common substring between two strings.
140. Count the number of distinct substrings in a string.
141. Split a string based on a delimiter without using built-in methods.
142. Find the length of the longest word in a string.
143. Find the smallest and largest words in a string.
144. Find the frequency of each word in a string.
145. Check if two strings differ by exactly one character.
146. Check if two strings are rotations of each other.
147. Check if a string contains another string as a subsequence.
148. Determine if two strings are k-anagrams.
149. Check if two strings are a valid shuffle of each other.

150. Check if a string is a valid number.
151. Check if a string is a valid email address.
152. Convert a string to its equivalent integer representation.
153. Implement your own substring() method.
154. Implement your own trim() method for strings.
155. Implement a basic string compression algorithm (e.g., "aabccccaaa" -> "a2b1c4a3").
156. Decode a run-length encoded string (e.g., "a2b3" -> "aabbb").
157. Implement your own split() method for a string.
158. Implement your own indexOf() method for strings.
159. Find the longest substring without repeating characters.
160. Find the longest common prefix among an array of strings.
161. Group anagrams from a list of strings.
162. Find all anagrams of a given word in a string.
163. Find the minimum window substring containing all characters of another string.
164. Split a string into balanced parts with equal 0s and 1s.
165. Check if a string is made up of unique characters.
166. Find the longest repeating substring in a string.
167. Reverse only the vowels in a string.
168. Check if a string can be rearranged to form a palindrome.