1. What is the time complexity of the Knuth-Morris-Pratt (KMP) algorithm for pattern searching in a text of length N and pattern of length M?
   1. O(N)
   2. O(M)
   3. **O(N+M)**
   4. O(N\*M)
2. The KMP algorithm is based on which principle?
   1. Divide and Conquer
   2. Greedy approach
   3. Sliding Window
   4. **Prefix-suffix matching**
3. In the Z function algorithm, the Z value of a character is defined as:
   1. **The length of the longest prefix of the text that matches the suffix starting at that character**
   2. The length of the longest suffix of the text that matches the prefix ending at that character
   3. The number of occurrences of the character in the text
   4. The number of occurrences of the character in the pattern
4. The Z function algorithm is used for:
   1. **Pattern searching**
   2. Pattern matching with wildcards
   3. Counting occurrences of a pattern in a text
   4. Longest common substring
5. The time complexity of the Z function algorithm for pattern searching in a text of length N and pattern of length M is:
   1. O(N)
   2. O(M)
   3. **O(N+M)**
   4. O(N\*M)
6. The Rabin-Karp algorithm is based on which principle?
   1. Prefix-suffix matching
   2. **Hashing**
   3. Divide and Conquer
   4. Dynamic Programming
7. The Rabin-Karp algorithm is primarily used for:
   1. **Exact pattern matching**
   2. Pattern matching with wildcards
   3. Pattern searching in sorted arrays
   4. Longest common substring
8. In the Rabin-Karp algorithm, the hash function is used to:
   1. Generate random numbers
   2. Calculate the ASCII value of characters in the pattern
   3. **Calculate the hash value of the pattern and the current window of the text**
   4. Calculate the factorial of a number
9. The time complexity of the Rabin-Karp algorithm for pattern searching in a text of length N and pattern of length M is:
   1. O(N)
   2. O(M)
   3. **O(N+M)**
   4. O(N\*M)
10. Which of the following is true about the Z function algorithm?
    1. It always performs better than the KMP algorithm for pattern searching.
    2. It is more efficient than the KMP algorithm for short texts.
    3. **It is based on dynamic programming principles.**
    4. It is a probabilistic algorithm.
11. The Z function algorithm is named after its inventor, Donald Zavala.
    1. True
    2. **False**
12. Which of the following is an advantage of the Rabin-Karp algorithm over the KMP algorithm?
    1. It has a better worst-case time complexity.
    2. **It works efficiently for large texts and patterns.**
    3. It does not use any additional data structures.
    4. It guarantees the shortest match for pattern searching.
13. The Z function algorithm works efficiently for pattern searching in a text of length N and pattern of length M, where M is:
    1. Greater than N
    2. Equal to N
    3. **Less than N**
    4. Any positive integer value
14. The KMP algorithm can be used for pattern matching with wildcards.
    1. True
    2. **False**
15. The Rabin-Karp algorithm has a worst-case time complexity of O(N\*M) for pattern searching.
    1. True
    2. **False**
16. In the KMP algorithm, the prefix function is used to:
    1. Calculate the hash value of the pattern and the current window of the text
    2. **Determine the length of the longest prefix that is also a suffix of the pattern**
    3. Divide the pattern into smaller parts for comparison
    4. Count the occurrences of the pattern in the text
17. The Z function algorithm is particularly efficient for:
    1. **Short texts and long patterns**
    2. Long texts and short patterns
    3. Texts with repeating characters
    4. Texts containing only lowercase letters
18. The time complexity of the Rabin-Karp algorithm becomes O(N) in the best-case scenario when:
    1. **The pattern occurs only once in the text**
    2. The hash function has a low collision rate
    3. The pattern and text are of the same length
    4. The pattern and text have the same characters
19. The KMP algorithm preprocesses the pattern to construct the longest prefix suffix (LPS) array in:
    1. O(N)
    2. **O(M)**
    3. O(N+M)
    4. O(N\*M)
20. The Rabin-Karp algorithm is not suitable for pattern matching with wildcards.
    1. **True**
    2. False
21. The Z function algorithm is not influenced by the alphabet size (number of characters in the text and pattern).
    1. **True**
    2. False
22. The Z function algorithm finds all occurrences of the pattern in the text in:
    1. O(N)
    2. O(M)
    3. **O(N+M)**
    4. O(N\*M)
23. The KMP algorithm is considered more efficient than the Rabin-Karp algorithm for large texts and patterns.
    1. True
    2. **False**
24. The Z function algorithm can be used to find the longest common prefix of two strings.
    1. **True**
    2. False
25. The Rabin-Karp algorithm uses modular arithmetic for hashing to avoid integer overflow.
    1. **True**
    2. False
26. The Z function algorithm finds the longest common prefix of a text with itself in:
    1. **O(N)**
    2. O(M)
    3. O(N+M)
    4. O(N\*M)
27. The KMP algorithm is particularly efficient for pattern searching in:
    1. **Texts with repeating patterns**
    2. Texts with large alphabets
    3. Texts with long patterns
    4. Texts with random characters
28. The Rabin-Karp algorithm is a deterministic algorithm, meaning it always guarantees the correct output for pattern searching.
    1. True
    2. **False**
29. The Z function algorithm can be used for approximate pattern matching.
    1. True
    2. **False**
30. The KMP algorithm is based on the observation that when a mismatch occurs, the pattern can be shifted by more than one position.
    1. True
    2. **False**
31. The Rabin-Karp algorithm performs well when the number of occurrences of the pattern in the text is very large.
    1. **True**
    2. False
32. The Z function algorithm can find all occurrences of a pattern in a text in:
    1. O(N)
    2. O(M)
    3. **O(N+M)**
    4. O(N\*M)
33. The KMP algorithm is not influenced by the order of characters in the pattern.
    1. **True**
    2. False
34. The Rabin-Karp algorithm can be used for pattern matching with regular expressions.
    1. True
    2. **False**
35. The Z function algorithm works efficiently for pattern searching in a text with repeating patterns.
    1. **True**
    2. False