1. In the sliding window technique, what does the window represent?
   1. **Subarray**
   2. Prefix sum
   3. Frequency array
   4. Binary search range
2. What is the advantage of using the sliding window technique?
   1. **It reduces the time complexity of the algorithm.**
   2. It guarantees an optimal solution for any problem.
   3. It simplifies the implementation of prefix arrays.
   4. It eliminates the need for auxiliary data structures.
3. Which algorithm uses the sliding window technique to find the longest subarray with a given sum?
   1. Kadane's algorithm
   2. Binary search algorithm
   3. **Two-pointer algorithm**
   4. Prefix sum algorithm
4. What is the time complexity of the sliding window technique?
   1. O(1)
   2. **O(n)**
   3. O(n log n)
   4. O(n^2)
5. Which of the following is NOT a step in the sliding window technique?
   1. Initialize window boundaries
   2. **Compute prefix sums**
   3. Update the window based on a condition
   4. Compute the maximum or minimum value in the window
6. Which problem can be efficiently solved using the sliding window technique?
   1. Finding the median of an array
   2. **Finding the number of distinct elements in a subarray**
   3. Sorting an array in non-decreasing order
   4. Finding the maximum prefix sum in an array
7. Which algorithm is used to find the smallest subarray with a sum greater than a given value?
   1. Kadane's algorithm
   2. Binary search algorithm
   3. **Two-pointer algorithm**
   4. Prefix sum algorithm
8. Which data structure is commonly used to implement a two-pointer algorithm?
   1. **Array**
   2. Linked list
   3. Stack
   4. Queue
9. What is the time complexity of the two-pointer algorithm?
   1. O(1)
   2. **O(n)**
   3. O(log n)
   4. O(n log n)
10. Which problem can be efficiently solved using the two-pointer algorithm?
    1. Finding the maximum subarray sum
    2. **Finding the longest palindromic substring**
    3. Finding the number of subarrays with a given sum
    4. Finding the smallest subarray with a sum greater than a given value
11. Which algorithm uses two pointers to find the longest palindromic substring?
    1. Kadane's algorithm
    2. Binary search algorithm
    3. **Two-pointer algorithm**
    4. Prefix sum algorithm
12. In the two-pointer algorithm, what do the two pointers represent?
    1. Subarray boundaries
    2. Prefix sum boundaries
    3. Frequency array boundaries
    4. **Indices in the array**
13. What is the advantage of using two pointers in algorithm design?
    1. **It reduces the time complexity of the algorithm.**
    2. It guarantees an optimal solution for any problem.
    3. It simplifies the implementation of prefix arrays.
    4. It eliminates the need for auxiliary data structures.
14. Which problem can be efficiently solved using two pointers?
    1. **Finding the maximum subarray sum**
    2. Finding the longest subarray with zero sum
    3. Finding the number of subarrays with a given sum
    4. Finding the smallest subarray with a sum greater than a given value
15. In the sliding window technique, what is the role of the two pointers?
    1. They define the size of the sliding window.
    2. They keep track of the maximum or minimum value in the window.
    3. **They determine when to expand or shrink the window.**
    4. They compute the prefix sums of the elements in the window.
16. Which problem can be efficiently solved using a frequency array?
    1. Finding the median of an array
    2. **Finding the number of distinct elements in a subarray**
    3. Sorting an array in non-decreasing order
    4. Finding the maximum prefix sum in an array
17. Which algorithm uses a frequency array to find the number of distinct elements in a subarray?
    1. Kadane's algorithm
    2. Binary search algorithm
    3. Two-pointer algorithm
    4. **Sliding window algorithm**
18. In the sliding window algorithm, how is the frequency array updated when the window moves?
    1. **The count of the outgoing element is decreased, and the count of the incoming element is increased.**
    2. The count of all elements in the window is recomputed from scratch.
    3. The frequency array is not updated during the sliding window algorithm.
    4. The frequency array is updated only when the window shrinks.
19. What is the time complexity of finding the smallest subarray with a sum greater than a given value using a sliding window and two pointers?
    1. O(1)
    2. **O(n)**
    3. O(log n)
    4. O(n log n)
20. Which data structure is commonly used to implement a sliding window?
    1. **Array**
    2. Linked list
    3. Stack
    4. Queue
21. Which technique is used to efficiently calculate prefix sums in an array?
22. Binary search
23. Sliding window
24. **Two pointers**
25. Dynamic programming
26. Which technique is typically used to update a frequency array while iterating over an array?
27. Two pointers
28. Binary search
29. **Sliding window**
30. Dynamic programming
31. Which technique is commonly used to find the longest subarray with a specific property using two pointers?
32. **Sliding window**
33. Binary search
34. Dynamic programming
35. Prefix array
36. What is the purpose of using two pointers in the sliding window technique?
37. **To maintain the window size.**
38. To iterate over the array elements.
39. To update the prefix array.
40. To keep track of subarray sums.
41. What is the main advantage of using the sliding window technique?
42. **It reduces time complexity by eliminating the need for nested loops.**
43. It allows for constant-time access to individual array elements.
44. It simplifies array manipulation by maintaining a fixed-size window.
45. It guarantees sorted subarrays in ascending order.
46. In the sliding window technique, how is the window size determined?
47. **It is fixed and specified in the problem statement.**
48. It is calculated based on the size of the input array.
49. It expands dynamically as needed during the computation.
50. It is determined randomly during the execution of the program.
51. What is the primary advantage of the sliding window technique over brute force approaches?
52. It allows for efficient processing of large datasets.
53. It guarantees the optimal solution for any problem.
54. **It reduces time complexity by eliminating unnecessary calculations.**
55. It eliminates the need for auxiliary data structures.
56. In the sliding window technique, when is the window expanded and when is it contracted?
57. The window is expanded when the sum of the current elements exceeds a given threshold and contracted when it falls below the threshold.
58. **The window is expanded when the current element satisfies a given condition and contracted when it no longer satisfies the condition.**
59. The window is expanded when the current element is larger than the previous element and contracted when it is smaller.
60. The window is expanded when the current element is smaller than the previous element and contracted when it is larger.
61. In the sliding window technique, when is the maximum length of the subarray with a specific property updated?
62. After the window is expanded.
63. **After the window is contracted.**
64. After each iteration of the loop.
65. After the entire array is processed.
66. Which of the following is NOT a step in implementing the sliding window technique?
67. Initialize the two pointers to the start of the array.
68. Determine the window size based on the problem requirements.
69. Expand or contract the window based on a specific condition.
70. **Repeat the process until the window covers the entire array.**
71. How can the sliding window technique be applied to find the longest substring with unique characters?
72. By expanding the window until a repeated character is found and contracting the window until no repeated characters are present.
73. **By contracting the window until a repeated character is found and expanding the window until no repeated characters are present.**
74. By expanding the window until no repeated characters are present and contracting the window until a repeated character is found.
75. By contracting the window until no repeated characters are present and expanding the window until a repeated character is found.