Western Governors University (WGU) C839v5 / D334 Algorithms Practice Test (Sample)

Study Guide

BY EXAMZIFY

Everything you need from our exam experts!

Questions

1. What characterizes a weighted graph?

A. All edges have values representing relationships

B. Each edge has a numerical value indicating cost or capacity

C. The graph contains no cycles

D. Each node must be connected to three other nodes

2. What is the key size and IV value for the lightweight symmetric stream cipher Enocoro?

A. 128-bit key size; 64-bit IV

B. 128-bit key size; 128-bit IV

C. 64-bit key size; 64-bit IV

D. 256-bit key size; 128-bit IV

3. What is an algorithm?

A. A method for random number generation

B. A logical puzzle to be solved

C. A step-by-step procedure or formula for solving a problem

D. A programming language syntax

4. What is the key size typically used for the Present cipher?

A. 64 or 256 bits

B. 80 or 128 bits

C. 128 or 192 bits

D. 128 or 2048 bits

5. What is the key size range for Twofish encryption?

A. 32-128 bits

B. 1-256 bits

C. 64-192 bits

D. 128-512 bits

6. What characterizes a greedy algorithm?

A. It considers all possible solutions before selecting one

B. It builds a solution incrementally with immediate benefits

C. It uses dynamic programming to solve problems

D. It works by exhaustively searching through all possibilities

7. What is the maximum key size that can be used in Blowfish encryption?

A. 256 bits

B. 448 bits

C. 512 bits

D. 128 bits

8. Which algorithmic approach is well suited for finding the shortest path in a graph?

A. Greedy algorithms

B. Backtracking

C. Dynamic programming

D. Divide and conquer

9. What is the principle behind Quick Sort?

A. It selects a 'pivot' and partitions the array based on it

B. It divides the array into two halves and sorts each half

C. It checks each element sequentially to find the target

D. It uses a hash table for quick look-up of elements

10. How does a merge sort algorithm primarily function?

A. By continuously merging data sets

B. By breaking down an unsorted list into smaller sublists

C. By swapping elements in place

D. By utilizing a single pivot value

Answers

1. B
2. A
3. C
4. B
5. B
6. B
7. B
8. C
9. A
10. B

Explanations

1. What characterizes a weighted graph?

A. All edges have values representing relationships

B. Each edge has a numerical value indicating cost or capacity

C. The graph contains no cycles

D. Each node must be connected to three other nodes

A weighted graph is characterized by each edge having a numerical value that represents an attribute of the edge, typically indicating cost or capacity. This numerical value allows for the representation of various practical scenarios such as distances between nodes, costs to traverse edges, or maximum flow rates. This feature is crucial in many algorithms used for graph analysis, such as Dijkstra's or Prim's algorithm, where these weights enable the determination of optimal paths or minimum spanning trees within the graph. In contrast, while it is true that all edges in a weighted graph may have values representing relationships, it is the specific numerical nature of those values indicating cost or capacity that solidifies its classification as a weighted graph. The definition excludes other graph characteristics like cycle presence or restrictions on node connectivity, as those pertain to different types of graphs, such as directed, undirected, acyclic, or specific connectivity conditions.

2. What is the key size and IV value for the lightweight symmetric stream cipher Enocoro?

A. 128-bit key size; 64-bit IV

B. 128-bit key size; 128-bit IV

C. 64-bit key size; 64-bit IV

D. 256-bit key size; 128-bit IV

The key size and IV (Initialization Vector) value for the lightweight symmetric stream cipher Enocoro is indeed a 128-bit key size with a 64-bit IV. This configuration is significant for a few reasons. The 128-bit key size allows for a substantial number of possible keys, providing a robust level of security against brute-force attacks. In encryption, a longer key size generally implies a greater difficulty for unauthorized parties to replicate the encryption process without the key. A 64-bit IV is used to ensure that even if the same plaintext is encrypted multiple times under the same key, the outputs will differ, thus enhancing security through randomness in the output. Using a 64-bit IV fits well within established practices for lightweight encryption solutions, where constraints on processing resources and performance can be critical. This cipher is designed for environments where efficiency is paramount, and the chosen sizes reflect a balance between security needs and operational feasibility.

3. What is an algorithm?

A. A method for random number generation

B. A logical puzzle to be solved

C. A step-by-step procedure or formula for solving a problem

D. A programming language syntax

An algorithm is defined as a step-by-step procedure or formula for solving a problem. This definition encompasses a series of well-defined instructions or rules that can be followed to achieve a specific result or solution. Algorithms are foundational in computer science and mathematics because they provide a clear method for executing tasks, performing calculations, or making decisions based on input data. For example, a simple algorithm for sorting a list of numbers might involve repeatedly comparing pairs of adjacent numbers and swapping them if they are in the wrong order. This systematic approach ensures that the list will be sorted by the end of the process. The clarity and structure of algorithms make them essential for designing computer programs, as they guide the logic used in coding and troubleshooting. The step-by-step nature of algorithms allows programmers to break down complex problems into manageable parts, ultimately leading to more efficient solutions.

4. What is the key size typically used for the Present cipher?

A. 64 or 256 bits

B. 80 or 128 bits

C. 128 or 192 bits

D. 128 or 2048 bits

The Present cipher is a lightweight block cipher that is specifically designed for use in resource-constrained environments, such as RFID tags and IoT devices. The key sizes used in the Present cipher are typically 80 bits or 128 bits. Choosing an 80-bit key allows for a balance between security and efficiency, making it suitable for applications where computational resources are limited. A 128-bit key provides a higher level of security, which is important for more sensitive applications, while still maintaining low resource requirements. Thus, the correct answer reflects the two common key sizes associated with the Present cipher effectively, emphasizing its design for security in minimal-resource situations.

5. What is the key size range for Twofish encryption?

A. 32-128 bits

B. 1-256 bits

C. 64-192 bits

D. 128-512 bits

The key size range for Twofish encryption is indeed from 128 bits to 256 bits. Twofish is a symmetric key block cipher that supports various key sizes, including 128, 192, and 256 bits. This flexibility allows Twofish to provide a good balance between security and performance, making it adaptable for different applications where varying levels of encryption strength might be needed. By supporting key sizes up to 256 bits, Twofish offers a robust security level that can withstand brute-force attacks, which enhances its utility in cryptographic applications requiring high security. The ability to use 128, 192, or 256 bits gives developers and system architects the choice to tailor their security posture based on their specific needs and threat models.

6. What characterizes a greedy algorithm?

A. It considers all possible solutions before selecting one

B. It builds a solution incrementally with immediate benefits

C. It uses dynamic programming to solve problems

D. It works by exhaustively searching through all possibilities

A greedy algorithm is characterized by its approach of building a solution incrementally, focusing on making the most advantageous local choice at each step with the hope that these local optimizations will lead to a global optimal solution. This means it selects the option that seems the best at the moment—often the one that offers the greatest immediate benefit or the most favorable outcome—without considering the broader implications or future consequences of that choice. It contrasts with other problem-solving strategies such as exhaustive search, which evaluates every possible solution, or dynamic programming, which solves complex problems by breaking them down into simpler subproblems and storing their results to avoid redundant calculations. Greedy algorithms are often simpler and more efficient than these other methods, but they do not guarantee a globally optimal solution for all problems; they work best on certain types of problems where local optimal choices lead to a global optimum.

7. What is the maximum key size that can be used in Blowfish encryption?

A. 256 bits

B. 448 bits

C. 512 bits

D. 128 bits

Blowfish encryption supports variable key sizes ranging from a minimum of 32 bits to a maximum of 448 bits. This flexibility allows users to choose a key size that meets their security needs while also balancing performance considerations. The 448 bits represent a very strong level of security, making it suitable for sensitive data encryption. Keys longer than 448 bits are not supported by the Blowfish algorithm, as the design and implementation are optimized for that key size limit. Therefore, the correct answer reflects the highest theoretically usable key length in Blowfish encryption as determined by its design specifications.

8. Which algorithmic approach is well suited for finding the shortest path in a graph?

A. Greedy algorithms

B. Backtracking

C. Dynamic programming

D. Divide and conquer

The dynamic programming approach is highly effective for finding the shortest path in a graph, particularly when using algorithms like Dijkstra's algorithm or the Bellman-Ford algorithm. This method systematically breaks down the problem into smaller subproblems and solves each of them efficiently. In the context of finding the shortest path, dynamic programming allows the algorithm to store and reuse results of previously computed paths, which significantly reduces unnecessary computations, especially in graphs with overlapping subproblems. For example, once the shortest path to a node is computed, that result can be utilized to compute paths to other nodes, avoiding the need to recalculate from scratch. Dynamic programming is particularly useful for graphs with certain characteristics, such as weighted edges. It excels in scenarios where the shortest path must be determined while considering varying weights, and the overlapping subproblem nature leads to efficient algorithm design. In contrast, greedy algorithms may work for certain types of graphs but do not always guarantee the shortest path solution due to their local optimization strategy. Backtracking is more suited for solving problems where all possibilities need to be explored and is less efficient for shortest path scenarios. Divide and conquer breaks problems into independent components, which does not align well with the needs of shortest path problems where dependencies exist between nodes.

9. What is the principle behind Quick Sort?

A. It selects a 'pivot' and partitions the array based on it

B. It divides the array into two halves and sorts each half

C. It checks each element sequentially to find the target

D. It uses a hash table for quick look-up of elements

The principle behind Quick Sort is centered on the selection of a 'pivot' element from the array and then partitioning the array based on that pivot. In this sorting algorithm, the elements are rearranged so that those less than the pivot come before it and those greater come after it. This process of partitioning is recursive, as the Quick Sort algorithm is applied to the sub-arrays formed by splitting at the pivot. This enables an efficient average-case time complexity of O(n log n). This method contrasts with other sorting techniques. For instance, some algorithms divide the array into two halves and sort them separately, which is typical of Merge Sort rather than Quick Sort. Additionally, checking each element sequentially to find a target refers to a linear search method, not a sorting method. Utilizing a hash table for quick look-ups does not pertain to sorting but rather to efficient data retrieval and access. Therefore, the essence of Quick Sort primarily lies in its pivot-based partitioning approach, which is captured in the correct answer.

10. How does a merge sort algorithm primarily function?

A. By continuously merging data sets

B. By breaking down an unsorted list into smaller sublists

C. By swapping elements in place

D. By utilizing a single pivot value

The merge sort algorithm primarily functions by breaking down an unsorted list into smaller sublists. This divide-and-conquer strategy involves recursively dividing the main list into halves until each sublist consists of a single element. Since a single element is inherently sorted, the algorithm then proceeds to merge these sublists back together in a manner that results in a fully sorted list. The critical aspect of merge sort is its systematic approach to sorting: first, the unsorted list is divided into smaller, more manageable parts, and then these parts are merged in a sorted order. This is where the efficiency of merge sort shines, particularly with larger datasets, as it consistently operates with a time complexity of O(n log n). While other options mention aspects of data manipulation or alternative sorting techniques, only the breakdown of the list into smaller components captures the essence of how merge sort operates. Without this division, effective merging would not be possible, which is why this strategy is foundational to the merge sort process.