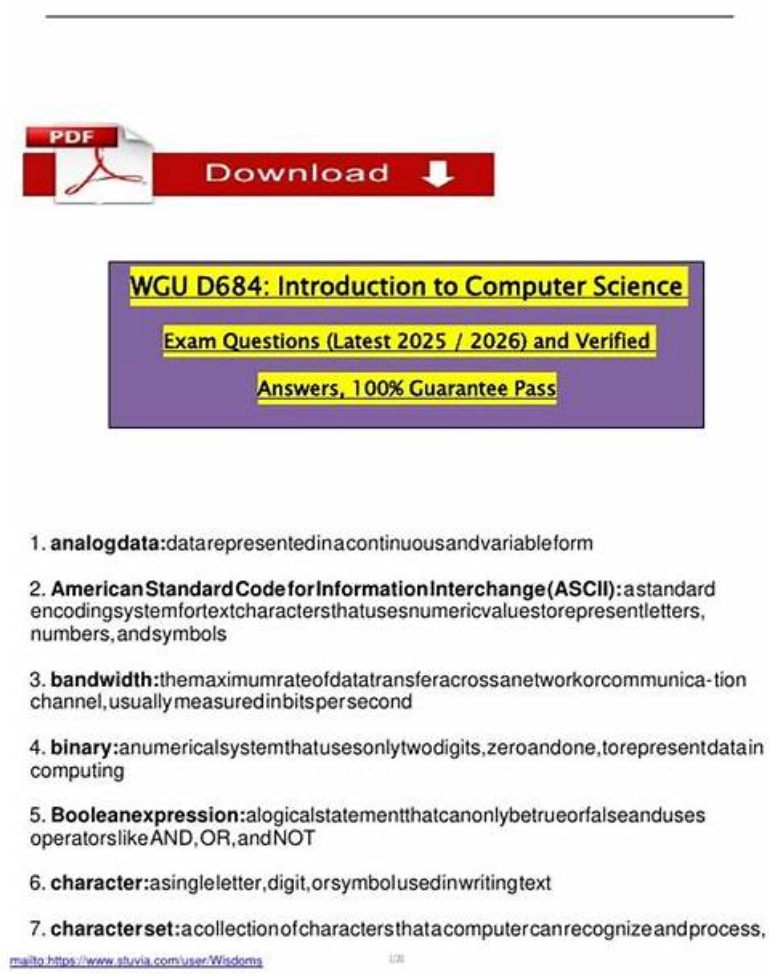
How Can WGU Foundations-of-Computer-Science Exam Questions Help You in Exam Preparation?



P.S. Free & New Foundations-of-Computer-Science dumps are available on Google Drive shared by VCE4Dumps: https://drive.google.com/open?id=1wXUssay-G3IsRbULt6YxP8rhWIiYwgak

This will help them polish their skills and clear all their doubts. Also, you must note down your WGU Foundations of Computer Science (Foundations-of-Computer-Science) practice test score every time you try the WGU Exam Questions. It will help you keep a record of your study and how well you are doing in them. VCE4Dumps hires the top industry experts to draft the WGU Foundations of Computer Science (Foundations-of-Computer-Science) exam dumps and help the candidates to clear their WGU Foundations of Computer Science (Foundations-of-Computer-Science) exam easily. VCE4Dumps plays a vital role in their journey to get the Foundations-of-Computer-Science certification.

VCE4Dumps's Foundations-of-Computer-Science exam certification training materials include Foundations-of-Computer-Science exam dumps and answers. The data is worked out by our experienced team and IT professionals through their own exploration and continuous practice, and its authority is unquestioned. You can download Foundations-of-Computer-Science free demo and answers on probation on VCE4Dumps website. After you purchase Foundations-of-Computer-Science exam certification training information, we will provide one year free renewal service.

>> Foundations-of-Computer-Science Valid Test Sims <<

Braindump Foundations-of-Computer-Science Free - Foundations-of-

Computer-Science Exam Pass Guide

We can say that how many the Foundations-of-Computer-Science certifications you get and obtain qualification certificates, to some extent determines your future employment and development, as a result, the Foundations-of-Computer-Science exam guide is committed to helping you become a competitive workforce, let you have no trouble back at home. Actually, just think of our Foundations-of-Computer-Science Test Prep as the best way to pass the exam is myopic. They can not only achieve this, but ingeniously help you remember more content at the same time.

WGU Foundations of Computer Science Sample Questions (Q49-Q54):

NEW QUESTION # 49
What will the expression fam[3:6] return?

- A. A list with elements at index 6
- B. A list with elements at index 3, 4, 5, and 6
- C. A list with elements at index 4, 5, and 6
- D. A list with elements at index 3, 4, and 5

Answer: D

Explanation:
Python slicing follows the rule `sequence[start:stop]`, where the `start` index is **inclusive** and the `stop` index is **exclusive**. This convention is taught widely because it makes many algorithms and boundary cases simpler: the length of the slice is `stop - start` (when step is 1), and adjacent slices can partition a sequence without overlap. For a list named `fam`, the slice `fam[3:6]` starts at index 3 and includes the elements at indices 3, 4, and 5, but it stops before index 6.
This is a frequent source of off-by-one errors for beginners, so textbooks emphasize remembering: "start is included, stop is not." If `fam` had at least 6 elements, then `fam[3:6]` would produce a new list of exactly three elements (positions 3, 4, 5). If `fam` had fewer than 6 elements, Python would still return a valid slice up to the end without raising an error, because slicing is designed to be safe within bounds.
Option A is incorrect because it skips index 3 and incorrectly includes index 6. Option B is incorrect because it includes index 6, which the stop boundary excludes. Option D is incorrect because slicing returns a sublist, not a single element; a single element would require indexing like `fam[6]`.

NEW QUESTION # 50
Which file system is commonly used in Windows and supports file permissions?

- A. FAT32
- B. NTFS
- C. EXT4
- D. HFS+

Answer: B

Explanation:
Windows commonly uses the NTFS (New Technology File System) for internal drives and many external drives because it supports advanced features required for modern operating systems. One of the most important features is support forfile and folder permissionsvia Access Control Lists (ACLs). Permissions enable the OS to enforce security policies by controlling which users and groups can read, write, execute, modify, or delete specific resources. This is fundamental to multi-user security and is a standard topic in operating systems and security textbooks.
FAT32 is an older file system designed for simplicity and broad compatibility. It does not provide the same fine-grained permission model as NTFS, which is why it is often used for removable media where cross- platform compatibility matters more than access control. HFS+ is historically associated with Apple's macOS systems, and EXT4 is widely used on Linux. While these file systems have their own permission and feature models, they are not the common Windows default for permission-managed storage in typical Windows deployments.
NTFS also supports journaling (improving reliability after crashes), large file sizes, quotas, compression, and encryption features (through Windows facilities). In enterprise environments, NTFS permissions integrate with Windows authentication and directory services, enabling centralized user management. Therefore, for Windows systems requiring file permissions, NTFS is the correct answer.

NEW QUESTION # 51
The np_2d array stores information about multiple family members. Each row represents a different person, and the columns store family member attributes in the following order:
Age (years)
Weight (pounds)
Height (inches)
How is the weight of all family members selected from the np_2d array?

- A. np_2d[:, 1]
- B. np_2d[2, :]
- C. np_2d[1, :]
- D. np_2d[:, 2]

Answer: A

Explanation:
In a 2D NumPy array, rows and columns represent different dimensions of the data. The indexing form array [row_selection, column_selection] allows you to select entire rows, entire columns, or submatrices. The slice : means "all indices along this dimension." Since each row corresponds to a family member (a person), selecting weights forallfamily members means selectingall rowsfor the weight column.
The problem states the columns are ordered as: Age (column 0), Weight (column 1), Height (column 2).
Therefore, the weight column has index 1. The expression np_2d[:, 1] uses : to take every row and 1 to take the second column, producing a 1D array (or a column view) containing the weight values for all people.
Option A, np_2d[:, 2], would select the height column, not weight. Option C, np_2d[2, :], selects the third row (the third person) and all columns-age, weight, and height for just that one person. Option D, np_2d[1, :], selects the second person's entire row.
This column selection technique is fundamental in data analysis because datasets are often stored as "rows = observations, columns = features," and extracting a feature vector is a frequent operation before computing statistics or building models.

NEW QUESTION # 52
Which action is taken if the first number is the lowest value in a selection sort?

- A. The first number is increased by one.
- B. It swaps the selected element with the last unsorted element.
- C. The first number is duplicated.
- D. It swaps the selected element with the first unsorted element.

Answer: D

Explanation:
Selection sort works by maintaining a boundary between a sorted prefix and an unsorted suffix. On each pass, the algorithm finds the smallest value in the unsorted portion and places it into the first position of that unsorted portion (which is also the next position in the sorted prefix). This is usually done by swapping the element at the minimum's index with the element at the boundary index (the "first unsorted element"). That description matches option D.
If the first element of the unsorted portion is already the smallest, then the minimum's index equals the boundary index. In textbook implementations, the algorithm may still execute a swap operation, but it becomes a swap of an element with itself (a no-op), leaving the array unchanged. Many implementations include a small optimization: perform the swap only if the minimum index differs from the boundary index.
Either way, conceptually the "action taken" by selection sort is still "swap the selected minimum into the first unsorted position," which is exactly what option D states.
Options A and B are unrelated to sorting; selection sort never increases or duplicates values. Option C is incorrect because selection sort swaps the minimum with thefirstunsorted element, not the last. After the swap (or no-op), the sorted region grows by one element, and the algorithm repeats from the next boundary position.
This logic is fundamental for understanding how selection sort ensures correctness: after pass i, the smallest i+1 elements are fixed in their final positions.

NEW QUESTION # 53
What is the time complexity of a quicksort algorithm?

- A. O(n log n)
- B. O(1)
- C. O(log n)
- D. O(n)

Answer: A

Explanation:
Quicksort is a divide-and-conquer sorting algorithm. It works by selecting a pivot element, partitioning the array into two subarrays (elements less than the pivot and elements greater than the pivot), and then recursively sorting those subarrays. In the average case, the partition step splits the array into roughly equal halves, so the recurrence is commonly written as (T(n) = T(n/2) + T(n/2) + O(n)), where (O(n)) is the cost of partitioning. This solves to (O(n \log n)), which is why quicksort is widely taught as an efficient general-purpose sorting method.
However, textbooks also emphasize that quicksort has a worst-case time complexity of (O(n