

Dumpcollection Is the Most Reliable Platform for WGU Foundations-of-Computer-Science Exam Preparation



What's more, part of that Dumpcollection Foundations-of-Computer-Science dumps now are free: <https://drive.google.com/open?id=1QgDXejjROiHBv4yh9mZiYPUWBGaG6URh>

Users don't need to install any plugins or software to attempt the WGU Foundations-of-Computer-Science practice exam. All operating systems support this format. The third and last format is WGU Foundations of Computer Science Foundations-of-Computer-Science desktop software that can be used on Windows computers. The customers that have Windows laptops or computers can attempt the practice exam and prepare for it efficiently. These formats are in use by a lot of applicants currently and they are preparing for their best future on daily basis. Even the customers who have used it in the past for the preparation of WGU Foundations-of-Computer-Science Certification Exam have rated our product as one of the best.

There are three versions of our Foundations-of-Computer-Science exam questions. And all of the PDF version, online engine and windows software of the Foundations-of-Computer-Science study guide will be tested for many times. Although it is not easy to solve all technology problems, we have excellent experts who never stop trying. And whenever our customers have any problems on our Foundations-of-Computer-Science Practice Engine, our experts will help them solve them at the first time.

>> Foundations-of-Computer-Science Pdf Demo Download <<

WGU Foundations-of-Computer-Science Pdf Questions - Outstanding Practice To your WGU Foundations of Computer Science Exam

If you are going to prepare for the Foundations-of-Computer-Science exam in order to get the related certification and improve yourself, you are bound to be very luck. With the joint efforts of all parties, our company has designed the very convenient and useful Foundations-of-Computer-Science study materials. More importantly, the practices have proven that the study materials from our company have helped a lot of people achieve their goal and get the related certification. The Foundations-of-Computer-Science Study Materials of our company is the study tool which best suits these people who long to pass the Foundations-of-Computer-Science exam and get the related certification.

WGU Foundations of Computer Science Sample Questions (Q40-Q45):

NEW QUESTION # 40

What is the expected output of `numpy_array[1]`?

- A. The second element of the array
- B. The first element of the array
- C. An error message in the array
- D. A display of the entire array

Answer: A

Explanation:

In Python and NumPy, indexing is zero-based, meaning the first element of a 1D sequence is at index 0, the second element is at index 1, and so on. A NumPy array behaves like a sequence for basic indexing, so `numpy_array[1]` returns the element stored at position 1 in the array. This is a fundamental concept taught in introductory programming and scientific computing: indexing selects a single element, while slicing selects a range.

For example, if `numpy_array = np.array([5, 8, 13])`, then `numpy_array[0]` is 5, `numpy_array[1]` is 8, and `numpy_array[2]` is 13. The expression `numpy_array[1]` therefore evaluates to the second element (8 in this example). This does not display the entire array (that would happen with `print(numpy_array)`), and it does not produce an error unless the array is too short. An error such as `IndexError` occurs only if index 1 is out of bounds, for example when the array has length 1 and you try to access `numpy_array[1]`.

Textbooks emphasize careful reasoning about indices because off-by-one errors are common. In data analysis, correct indexing is crucial for extracting the right observations, features, or time steps from numerical datasets.

NEW QUESTION # 41

What is the first step in the selection sort algorithm?

- A. Swap the first and last elements.
- B. Find the highest value and the lowest value in the list.
- C. Determine the lowest value starting from the first position.
- D. Sort the list in descending order.

Answer: C

Explanation:

Selection sort works by growing a sorted portion of the list one element at a time. The algorithm conceptually divides the array into two regions: a sorted prefix on the left and an unsorted suffix on the right. At the beginning, the sorted prefix is empty and the entire list is unsorted. The first step is to consider position 0 as the target location for the smallest element. The algorithm scans the unsorted region (initially the whole list) to find the smallest value and records its index. That action is exactly what option C describes: determine the lowest value starting from the first position.

After identifying the minimum element, selection sort swaps it into position 0 (if it isn't already there). Then it repeats the process for position 1, scanning the remaining unsorted suffix to find the next smallest element, swapping it into place, and so on. Textbooks emphasize that the key characteristic of selection sort is the repeated "select min (or max) from unsorted region and place it into the sorted region." Option A is not the standard first step; finding both min and max is unnecessary. Option B describes an unrelated swap that doesn't ensure progress toward sorting. Option D is not a "first step" but rather a different ordering goal; selection sort can be adapted for descending order, but the canonical version begins by selecting the minimum for the first position.

NEW QUESTION # 42

What is the method for changing an element in a Python list?

- A. Use square brackets and the equals sign
- B. Use parentheses and the plus sign
- C. Use the `del` keyword and the element's value
- D. Use curly brackets and the equals sign

Answer: A

Explanation:

In Python, a list is a mutable sequence, meaning its elements can be changed after the list is created. The standard textbook method for updating a specific element is index assignment, which uses square brackets to select the position and the equals sign to assign a new value. For example, if `nums = [10, 20, 30]`, then `nums[1] = 99` changes the element at index 1 from 20 to 99, producing `[10, 99, 30]`. This works because lists store references to objects and allow those references to be updated in-place.

Option B is incorrect because parentheses are used for function calls and tuples, and the plus sign typically performs concatenation (creating a new list) rather than modifying an existing element by position. Option C is incorrect because curly brackets denote dictionaries or sets, not lists. Option D is incorrect because `del` removes elements by index or slice (for example, `del nums[1]`), and it does not delete by "the element's value" unless you first find the index. Deleting is not the same as changing; deletion reduces the list's length and shifts later indices.

Index assignment is fundamental in list manipulation and appears in standard algorithms: updating counters, replacing sentinel values, editing collections, and implementing in-place transformations efficiently without allocating a new list.

NEW QUESTION # 43

Which method allows a user to convert a string value to all capital letters in Python?

- A. makeUpper()
- B. toUpperCase()
- C. upperCase()
- D. upper()

Answer: D

Explanation:

In Python, strings are objects of type `str`, and the language provides many built-in string methods for common transformations. The standard method used to convert all alphabetic characters in a string to uppercase is `upper()`. For example, `"Hello, World".upper()` produces `"HELLO, WORLD"`. This method is part of Python's core string API and is documented as returning a new string because strings are immutable in Python; the original string is not modified.

Options A and D resemble methods from other programming languages. For instance, `toUpperCase()` is commonly seen in Java and JavaScript, not Python. Option B, `makeUpper()`, is not a standard method in Python's `str` type. Python's naming conventions for built-in methods are typically short and lowercase, which is consistent with `upper()`, `lower()`, `strip()`, and `replace()`.

It is also important to note what `upper()` does and does not do. It affects letters according to Unicode case-mapping rules, so it works beyond ASCII and supports many languages. Non-alphabetic characters such as digits, punctuation, and whitespace remain unchanged. Because the method returns a new string, it supports functional-style programming and safe reuse of the original data. In many textbook examples, `upper()` is paired with input normalization tasks, such as case-insensitive comparisons and cleaning user-entered text.

NEW QUESTION # 44

Which statement describes the relationship between trees and graphs?

- A. Trees cannot have cycles.
- B. Trees can have unconnected nodes.
- C. Trees can have cycles.
- D. Trees do not have levels.

Answer: A

Explanation:

In discrete mathematics and computer science, a tree is a special kind of graph. The standard graph-theory definition is that a tree is a connected, acyclic undirected graph. "Acyclic" means it contains no cycles, i.e., you cannot start at a vertex, follow a sequence of edges, and return to the starting vertex without repeating edges in a way that forms a loop. (Wikipedia) This property is exactly what makes option D correct.

The other options contradict the definition. If a structure has cycles, it is not a tree (though it may still be a graph). If it has unconnected nodes, it is not connected; such a structure is more like a forest (a disjoint union of trees) rather than a single tree.

(Wikipedia) The idea of "levels" belongs to a particular computer-science representation called a rooted tree, where one node is chosen as the root and nodes can be assigned depths

/levels based on distance from the root. But levels are not required in the abstract definition of a tree as a graph; they arise from choosing a root and orientation for convenience in algorithms like BFS/DFS, heaps, and parse trees.

So, the relationship is: every tree is a graph with extra structure—specifically, no cycles and (typically) connectivity—and the "no cycles" rule is the key distinguishing feature. (Discrete Mathematics)

NEW QUESTION # 45

.....

A WGU Foundations of Computer Science will not only expand your knowledge but it will polish your abilities as well to advance successfully in the world of WGU. Real WGU Foundations-of-Computer-Science Exam QUESTIONS certification increases your commitment and professionalism by giving you all the knowledge necessary to work in a professional setting. We have heard from thousands of people who say that using the authentic and Reliable Foundations-of-Computer-Science Exam Dumps was the only way they were able to pass the Foundations-of-Computer-Science.

New Foundations-of-Computer-Science Exam Simulator: https://www.dumpcollection.com/Foundations-of-Computer-Science_braindumps.html

P.S. Free 2026 WGU Foundations-of-Computer-Science dumps are available on Google Drive shared by Dumpcollection:
<https://drive.google.com/open?id=1QgDXeijROiHBv4yh9mZiYPUWBGaG6URh>