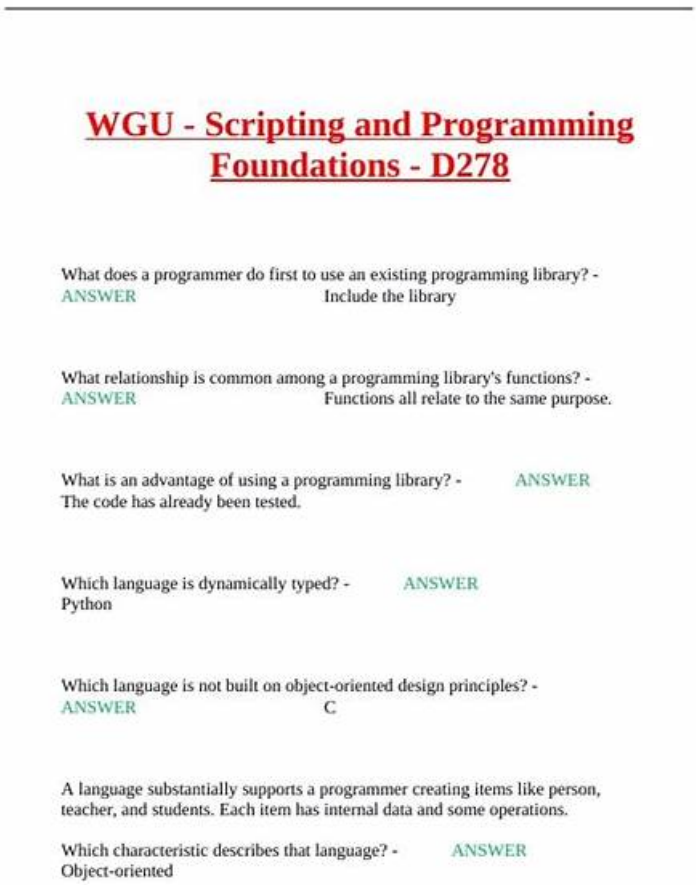
Scripting-and-Programming-Foundations試験の準備方法｜一番優秀なScripting-and-Programming-Foundations最新な問題集試験｜高品質なWGU Scripting and Programming Foundations Exam日本語関連対策


WGU - Scripting and Programming Foundations - D278

What does a programmer do first to use an existing programming library? - ANSWER Include the library

What relationship is common among a programming library's functions? - ANSWER Functions all relate to the same purpose.

What is an advantage of using a programming library? - ANSWER The code has already been tested.

Which language is dynamically typed? - ANSWER Python

Which language is not built on object-oriented design principles? - ANSWER C

A language substantially supports a programmer creating items like person, teacher, and students. Each item has internal data and some operations.

Which characteristic describes that language? - ANSWER Object-oriented

BONUS！！！ MogiExam Scripting-and-Programming-Foundationsダンプの一部を無料でダウンロード：https://drive.google.com/open?id=1nvycaRZQMVDrKuoLuQY-QDzk2AflyN0H

親愛なる受験生の皆様、何かWGUのScripting-and-Programming-Foundations試験のトレーニング授業に加入したいのですか。実は措置を取ったら一回で試験に合格することができます。MogiExamのWGUのScripting-and-Programming-Foundations試験のトレーニング資料はとても良い選択なんですよ。MogiExamの仮想ネットワークトレーニングと授業は大量の問題集に含まれていますから、ぜひあなたが気楽に試験に合格することを約束します。

WGU Scripting-and-Programming-Foundations 認定試験の出題範囲：

トピック	出題範囲

トピック 1	• Explaining Logic and Outcomes of Simple Algorithms: This section of the exam measures the skills of Entry Level Programmers and covers the ability to read simple algorithms and understand how they work. It focuses on predicting outputs, understanding step by step logic, and identifying how basic instructions create a final result. The goal is to help learners understand algorithm reasoning without requiring advanced coding knowledge.
トピック 2	• Using Fundamental Programming Elements: This section of the exam measures skills of Entry Level Programmers and covers the use of basic programming components required in everyday tasks. It includes working with variables, loops, conditions, and simple logic to perform common operations. The focus is on applying these elements correctly to complete small programming assignments in a clear and organized way.
トピック 3	• Identifying Scripts for Computer Program Requirements: This section of the exam measures the skills of Junior Software Developers and covers the ability to match a task with the correct script or programming approach. It highlights how different scripts can satisfy specific requirements and how to recognize the right structure for a given programming problem.
トピック 4	• Scripting and Programming Foundations: This section of the exam measures the skills of Junior Software Developers and covers the essential building blocks of programming. It focuses on variables, data types, flow control, and basic design concepts. Learners understand how programming logic works and how different languages handle similar tasks. The section also introduces the difference between interpreted and compiled languages in a simple and practical way.

Scripting-and-Programming-Foundations日本語関連対策、Scripting-and-Programming-Foundations問題集

MogiExam を選択して１００％のScripting-and-Programming-Foundations合格率を確保することができて、もしScripting-and-Programming-Foundations試験に失敗したら、MogiExamが全額で返金いたします。

WGU Scripting and Programming Foundations Exam 認定 Scripting-and-Programming-Foundations 試験問題 (Q68-Q73):

質問 # 68
The steps in an algorithm to build a picnic table are given.
1) Measure and mark the lumber cuts that need to be made
2) Buy the needed materials
3) Determine the needed materials
4) Cut the lumber to the proper dimensions
5) Assemble the pieces and paint.
Which two steps of the algorithm should be switched to make the algorithm successful?

- A. 1 and 3
- B. 1 and 2
- C. 2 and 3
- D. 2 and 4

正解：C

解説：
* Measure and mark the lumber cuts: This step involves measuring and marking the specific cuts required for the picnic table. It ensures that the lumber pieces are appropriately sized for assembly.
* Determine the needed materials: Before purchasing materials, it's essential to determine what is required. This step involves creating a list of necessary items such as lumber, screws, paint, etc.
* Buy the needed materials: Once the materials list is ready, proceed to purchase them. This step ensures that you have all the necessary supplies before starting the construction.

* Cut the lumber to the proper dimensions: With the materials on hand, cut the lumber according to the measurements marked in step 1. This ensures that the pieces fit together correctly during assembly.
* Assemble the pieces and paint: Finally, assemble the cut lumber pieces to create the picnic table. After assembly, apply paint or finish as desired.
References
* No specific references are provided for this question, but the steps align with general woodworking practices for constructing a picnic table. You can refer to woodworking guides or carpentry resources for further details.

質問 # 69
Which problem is solved by DijkStra's shortest path algorithm?

- A. Given an increasing array of numbers is the number 19 in the array?
- B. Given an alphabetized list of face entrants and a person's name, is the person entered in the race?
- C. Given two newspaper articles what is the greatest sequence of words shared by both articles?
- D. Given the coordinates of five positions, what is the most fuel-efficient flight pain?

正解：D

解説：
Dijkstra's shortest path algorithm is designed to find the shortest path between nodes in a graph. This can be applied to various scenarios, such as routing problems, network optimization, and in this case, determining the most fuel-efficient flight plan. The algorithm works by iteratively selecting the unvisited vertex with the smallest tentative distance from the source, then visiting the neighbors of this vertex and updating their tentative distances if a shorter path is found. This process continues until the destination vertex is reached or all reachable vertices have been visited.
In the context of the given options, Dijkstra's algorithm is best suited for option B, where the goal is to find the most fuel-efficient path (i.e., the shortest path) between multiple points (coordinates of five positions). The algorithm is not designed to solve problems like searching for an element in an array (option A), finding the longest common subsequence (option C), or searching for a name in a list (option D).
References:
* GeeksforGeeks article on Dijkstra's Algorithm1
* Wikipedia page on Dijkstra's Algorithm2
* Programiz explanation of Dijkstra's Algorithm3

質問 # 70
A programmer has been hired to create an inventory system for the books in a library. What is the waterfall phase in which waterfall outlining all the functions that need to be written to support the inventory system?

- A. Testing
- B. Implementation
- C. Design
- D. Analysis

正解：C

解説：
In the Waterfall model of software development, the phase where all functions that need to be written to support the inventory system would be outlined is the Design phase. This phase is critical as it translates the requirements gathered during the analysis phase into a blueprint for constructing the system. It involves two subphases: logical design and physical design. The logical design subphase is where possible solutions are brainstormed and theorized, while the physical design subphase is when those theoretical ideas and schemas are turned into concrete specifications12.

質問 # 71
Which output results from the given algorithm?
i = 61
d = 6
c = 0
while i >= d

c = c + 1
i = i - d
Put c to output

- A. 0
- B. 1
- C. 2
- D. 3

正解：B

解説：
Comprehensive and Detailed Explanation From Exact Extract:
The algorithm counts how many times d can be subtracted from i until i is less than d, effectively computing the integer division i / d.
According to foundational programming principles, we trace the loop execution.
* Initial State:
* i = 61, d = 6, c = 0.
* Loop Execution:
* While i >= d (i.e., 61 >= 6):
* c = c + 1 (increment counter).
* i = i - d (subtract d from i).
* Iterations:
* 1: i = 61, c = 0 + 1 = 1, i = 61 - 6 = 55.
* 2: i = 55, c = 1 + 1 = 2, i = 55 - 6 = 49.
* 3: i = 49, c = 2 + 1 = 3, i = 49 - 6 = 43.
* 4: i = 43, c = 3 + 1 = 4, i = 43 - 6 = 37.
* 5: i = 37, c = 4 + 1 = 5, i = 37 - 6 = 31.
* 6: i = 31, c = 5 + 1 = 6, i = 31 - 6 = 25.
* 7: i = 25, c = 6 + 1 = 7, i = 25 - 6 = 19.
* 8: i = 19, c = 7 + 1 = 8, i = 19 - 6 = 13.
* 9: i = 13, c = 8 + 1 = 9, i = 13 - 6 = 7.
* 10: i = 7, c = 9 + 1 = 10, i = 7 - 6 = 1.
* Stop: i = 1 < 6, exit loop.
* Output: Put c to output outputs c = 10.
* Verification: 61 ÷ 6 = 10 (integer division), with remainder 1 (since 61 - 6 * 10 = 1).
* Option A: "1." Incorrect. This would occur after one iteration.
* Option B: "5." Incorrect. This is too low; c reaches 10.
* Option C: "10." Correct. Matches the count of subtractions.
* Option D: "60." Incorrect. This is unrelated to the algorithm's logic.
Certiport Scripting and Programming Foundations Study Guide (Section on Loops and Counters).
Python Documentation: "While Loops" (https://docs.python.org/3/reference/compound_stmts.html#while).
W3Schools: "C While Loop" (https://www.w3schools.com/c/c_while_loop.php).

質問 # 72
One requirement for the language of a project is that it is based on a series of cells. Which type of language is characterized in this way?

- A. Markup
- B. Functional
- C. Compiled
- D. Static

正解：A

解説：
Comprehensive and Detailed Explanation From Exact Extract:
The term "based on a series of cells" is commonly associated with markup languages, particularly in the context of web development, where content is structured in a hierarchical or cell-based layout (e.g., HTML tables or CSS grid systems). According to foundational programming principles, markup languages like HTML are characterized by their use of tags to define elements, which can be visualized as cells or containers for content.

* Option A: "Functional." This is incorrect. Functional languages (e.g., Haskell, Lisp) focus on functions as first-class citizens and immutability, not on a cell-based structure.
* Option B: "Static." This is incorrect. "Static" refers to typing (where types are fixed at compile time) or analysis, not a cell-based structure.
* Option C: "Markup." This is correct. Markup languages like HTML use tags to create elements that can be arranged in a cell-like structure (e.g., <table> or elements in a grid). In web design, tables and CSS layouts are often described as cell-based.
* Option D: "Compiled." This is incorrect. Compiled languages (e.g., C, Java) are translated into machine code before execution, but they are not characterized by a cell-based structure.
Certiport Scripting and Programming Foundations Study Guide (Section on Markup Languages).
W3Schools: "HTML Tables" (https://www.w3schools.com/html/html_tables.asp).
Mozilla Developer Network: "CSS Grid Layout" (https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout).

質問 # 73
.....

Scripting-and-Programming-Foundations模擬試験の合格率はほぼ100％ですが、合格しない可能性がある場合は、全額返金することができます。払い戻しプロセスの複雑さを心配する必要はまったくありません。非常にシンプルにしています。 Scripting-and-Programming-Foundationsの使用後に試験に不合格になったことの証明を提供していただければ、すぐに返金できます。払い戻しプロセス中に問題が発生した場合は、いつでもカスタマーサービススタッフに連絡することもできます。問題をできるだけ早く解決するのに役立ちます。つまり、Scripting-and-Programming-Foundations試験問題は、試験に合格することをほぼ保証します。

Scripting-and-Programming-Foundations受験記対策 □□ www.mogiexam.com □で➥ Scripting-and-Programming-Foundations □を検索して、無料で簡単にダウンロードできますScripting-and-Programming-Foundations試験復習赤本

- bbs.t-firefly.com, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, stackblitz.com, pixabay.com, bbs.t-firefly.com, www.stes.tyc.edu.tw, bbs.t-firefly.com, www.stes.tyc.edu.tw, 121.199.46.216, Disposable vapes

P.S.MogiExamがGoogle Driveで共有している無料の2026 WGU Scripting-and-Programming-Foundationsダンプ：https://drive.google.com/open?id=1nvycaRZQMVDrKuoLuQY-QDzk2AflyN0H