Test bank of questions that can be used for review and/or assessment. Some questions are a review from previous units, and some are specifically based on Unit 4 topics.

|  |  |
| --- | --- |
| **AP CSP Unit 4 Vocabulary** | |
| Iterative process | 1. Repeatedly taking small steps to build a whole solution 2. Continuously adjusting the system to keep the error close to zero 3. Automates control of a system by comparing the output state to input 4. A marker placed on a line of code that causes the debugger to stop |
| List | 1. The number used to access an item in a list. 2. An individual value in a list. 3. An ordered collection of items. 4. The number of items in a list. |
| Index | 1. The number used to access an item in a list. 2. An individual value in a list. 3. An ordered collection of items. 4. The number of items in a list. |
| Element | 1. The number used to access an item in a list. 2. An individual value in a list. 3. An ordered collection of items. 4. The number of items in a list. |
| Length | 1. The number used to access an item in a list. 2. An individual value in a list. 3. An ordered collection of items. 4. The number of items in a list. |
| Global variable | 1. Specific numbers used in code that should be constants 2. Variables defined inside a function 3. Variables defined outside a function in the main program 4. A list of variables declared in a function definition that receive values |
| Local variable | 1. Specific numbers used in code that should be constants 2. Variables defined inside a function 3. Variables defined outside a function in the main program 4. A list of variables declared in a function definition that receive values |

|  |  |
| --- | --- |
| **AP CSP Unit 4 Coding Questions** | |
| What does this code do? | 1. Waits for a button press to go again. 2. Ends the code. 3. Remote control for the ‘bot. 4. Recalculates the distance. |
| How can you debounce the button? | 1. Hold the button for one second. 2. Add a sleep() delay. 3. Add another while loop to the code. 4. Call buttons.was\_pressed() |
| To make your code more readable, what section should go right after imports? | 1. Functions 2. Constants 3. Global variables 4. The main program |
| What is the data type of this value:  **5** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What is the data type of this value:  **(1, 0, 1, 0, 0)** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What is the data type of this value:  **4.3** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What is the data type of this value:  **‘Python’** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What is the data type of this value:  **False** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What is the data type of this value:  **[400, 500, 600]** | 1. Float 2. Boolean 3. Integer 4. String 5. List 6. Tuple |
| What line of code initializes, or defines, a counter variable? | 1. count = 0 2. count = count + 1 3. if count == 1: 4. def count = 0 |
| What line of code increments a counter? | 1. count = 0 2. count = count + 1 3. if count == 1: 4. def count = 0 |
| What line of code compares a counter to 1? | 1. count = 0 2. count = count + 1 3. if count == 1: 4. def count = 0 |
| The following code is an example of: | 1. Iteration 2. Selection 3. Sequencing 4. Randomization |
| The following code is an example of: | 1. Iteration 2. Selection 3. Sequencing 4. Randomization |
| The following code is an example of: | 1. Iteration 2. Selection 3. Sequencing 4. Randomization |
| What line of code defines an empty list? | 1. def new\_list 2. new\_list = { } 3. new\_list = ( ) 4. new\_list = [ ] |
| What code will add a new value to a list? | 1. my\_list.append(new\_value) 2. my\_list.add[new\_value] 3. append.new\_list(new\_value) 4. add.new\_list[new\_value] |
| What is the value of freq after the code snippet is run? | 1. 2 2. 600 3. 900 4. An error occurs |
| What is the final value of alist after this code segment is run? | 1. alist = [4, 0, 4, 8] 2. alist = [4, 0, 2, 4, 8] 3. alist = [0, 2, 5, 7] 4. alist = [4, 0, 2, 5, 7] |
| What is the index of the first element in a list? | 1. 0 2. 1 3. A 4. It depends on the elements in the list |
| What is the index of the last element in a list? | 1. len(the\_list) 2. len(the\_list) - 1 3. Z 4. It depends on the elements in the list |
| This is an example of: | 1. Accessing an element in a list 2. Accessing a random element in a list 3. Traversing a single list 4. Traversing multiple lists |
| This is an example of: | 1. Accessing an element in a list 2. Accessing a random element in a list 3. Traversing a single list 4. Traversing multiple lists |
| This is an example of: | 1. Accessing an element in a list 2. Accessing a random element in a list 3. Traversing a single list 4. Traversing multiple lists |
| This is an example of: | 1. Accessing an element in a list 2. Accessing a random element in a list 3. Traversing a single list 4. Traversing multiple lists |
| What is the correct function call for wait()? | 1. wait() 2. wait(delay) 3. def wait(): 4. done = wait() |
| What is the correct function call for my\_function()? | 1. my\_function() 2. my\_function(new\_list) 3. def my\_function(nlist): 4. answer = my\_function(new\_list) |
| In the function, what type of variable is **nlist**? | 1. Random 2. Global 3. Local 4. Parameter |
| In the function, what type of variable is **sum**? | 1. Random 2. Global 3. Local 4. Parameter |
| What does **my\_function()** do? | 1. Calculates the sum of a list 2. Finds the highest number in a list 3. Calculates the average of a list 4. An error occurs |
| What is the correct function call for does\_something()? | 1. do\_something() 2. do\_something(5) 3. def do\_something(thing): 4. call do\_something(thing) |
| In the function above, what type of variable is **found**? | 1. Random 2. Global 3. Local 4. Parameter |
| In the function above, what type of variable is **thing**? | 1. Random 2. Global 3. Local 4. Parameter |
| In the function above, what type of variable is **item**? | 1. Random 2. Global 3. Local 4. Parameter |
| What does **do\_something()** do? | 1. Searches a list for a specific item 2. Calculates the total of a list 3. Updates an element in a list 4. An error occurs |
| What happens if you don’t use this code in your function? | 1. Your code will crash 2. Nothing happens; the global line isn’t necessary 3. It will make a local variable into a global 4. It will make a global variable into a local |