

1. Which process helps identify the functions that should make up a computer program?
 1. black boxing
 2. stepwise refinement
 3. parameter passing
 4. debugging

Title

Which process helps identify the functions that should make up a computer program?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-01

from

testbank-py-1-ch05-01

2. The term Black Box is used with functions because
 1. Only the implementation matters; the specification is not important.
 2. Only the specification matters; the implementation is not important.
 3. Only the arguments matter; the return value is not important.
 4. Only the return value matters; the arguments are not important.

Title

Why is the term black box used with functions?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-02

from

testbank-py-1-ch05-02

3. One advantage of designing functions as black boxes is that
 1. many programmers can work on the same project without knowing the internal implementation details of functions.
 2. the result that is returned from black-box functions is always the same data type.
 3. the implementation of the function is open for everyone to see.
 4. there are fewer parameters.

Title

What is one advantage of designing functions as black boxes?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-03

from

testbank-py-1-ch05-03

4. A _____ is a sequence of instructions with a name.
1. variable
 2. argument
 3. parameter
 4. function

Title

What is a sequence of instructions with a name called in Python?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-04

5. What is supplied to a function when it is called?
1. arguments
 2. numbers
 3. return values
 4. variables

Title

What is supplied to a function when it is called?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-05

from

testbank-py-1-ch05-05

6. Consider the following function call `round(3.14159, 3)` what is the return value?
1. 3.14159
 2. 3.141
 3. 3.14
 4. 3.1

Title

What is returned from the round function when it is called?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-06

7. Consider the following function call `ceil(3.14159)` what is the return value?
1. 3.14159
 2. 3.0

3. 4.0
4. 3.1416

Title

What is returned from the ceil function when it is called?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-07

8. Which of the following is a correct call to Python's round function?

1. `x = round("3.14159", 2)`
2. `x = round("3.14159")`
3. `x = round(3.14159)`
4. `x = round(3, 1, 4, 1, 5, 9)`

Title

Which of the following is a correct function call for Python's round function?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-08

9. Consider a function named `calc`. It accepts two integer arguments and returns their sum as an integer. Which of the following statements is a correct invocation of the `calc` function?

1. `total = calc()`
2. `total = calc(2)`
3. `total = calc("2", "3")`
4. `total = calc(2, 3)`

Title

Which of the following is a correct function call?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-09

10. Which of the following statements correctly defines a function?

1. `def functionName(parameterName1, parameterName2) :`
2. `def functionName(parameterName1, parameterName2)`
3. `functionName(parameterName1, parameterName2) :`
4. `functionName(parameterName1, parameterName2)`

Title

Which of the following statements correctly defines a function?

type

mc
 Section
 5.2 Implementing and Testing Functions
 id
 testbank-py-2-ch05-10
 from
 testbank-py-1-ch05-10

11. What Python statement exits a function and gives the result to the caller?

1. def
2. return
3. send
4. result

Title
 What Python statement exits the function and gives the result to the caller?
 type
 mc
 Section
 5.2 Implementing and Testing Functions
 id
 testbank-py-1-ch05-11

12. Given the code snippet below, what is returned by the function call: `mystery(5, 3)`?

```
def mystery(num1, num2) :
    result = num1 * num2
    return result
```

1. 8
2. 15
3. 2
4. 0

Title
 What is the result of the given function call?
 type
 mc
 Section
 5.2 Implementing and Testing Functions
 id
 testbank-py-1-ch05-12

13. Given the code snippet below, what is returned by the function call: `mystery(mystery(5, 3), mystery(5, 3))`?

```
def mystery(num1, num2) :
    result = num1 * num2
    return result
```

1. 225
2. 15
3. 30

4. 0

Title

What is the result of the given function call?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-13

14. What is wrong with the following code snippet?

```
mystery(10, 2)
def mystery(num1, num2) :
    result = num1 ** num2
    return result
```

1. nothing, it will return 20
2. nothing, it will return 100
3. a variable must be used to store the result of the function call
4. the function must be defined before the statement that calls it

Title

What is wrong with the following code snippet?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-14

15. Consider the following function:

```
def w(x, y) :
    z = x + y
    return z
```

What is the function's name?

1. w
2. x
3. y
4. z

Title

What is the function's name?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-15

16. The following function is supposed to compute the area of a triangle and return the area as the function's result.

```
def triangleArea(base, height) :
    area = base * height / 2
```

What line of code must be placed in the blank to achieve this goal?

1. print area
2. print(area)
3. return area
4. return triangleArea

Title

What line of code is needed to complete the function for computing the area of a triangle?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-16

17. Assume that you are writing a function that computes the volume of a box for shipping electrical components. The components vary in shape -- some are long and skinny, while others are cube-like. Different boxes are used for components with different shapes. Which of the following function headers is the best?

1. def boxVolume() :
2. def boxVolume(sideLength) :
3. def boxVolume(a, b, c) :
4. def boxVolume(length, width, height) :

Title

Select the best header for a function that computes the volume of a box

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-17

18. Consider the following function.

```
def factorial(n) :
    result = 1
    for i in range(1, n + 1) :
        result = result * i
    return result
```

What is the parameter variable for this function?

1. factorial
2. i
3. n
4. result

Title
Identify the parameter variable for a function
type
mc
Section
5.2 Implementing and Testing Functions
id
testbank-py-2-ch05-18
from
testbank-py-1-ch05-18

19. Consider the following function:

```
def squareArea(sideLength) :  
    return sideLength ** 2
```

What is the value of squareArea(3)?

1. 2
2. 3
3. 6
4. 9

Title
Trace a function call
type
mc
Section
5.2 Implementing and Testing Functions
id
testbank-py-1-ch05-19

20. Consider the following function:

```
def squareArea(sideLength) :  
    return sideLength ** 2
```

What is the value of squareArea(squareArea(2))?

1. 2
2. 4
3. 8
4. 16

Title
Trace a nested function call
type
mc
Section
5.2 Implementing and Testing Functions
id
testbank-py-1-ch05-20

21. Consider the following function:

```
def mystery(a, b) :
    result = (a - b) * (a + b)
    return result
```

What is the result of calling `mystery(3, 2)`?

1. 2
2. 3
3. 4
4. 5

Title

Trace a function call

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-21

22. Consider the following function:

```
## Compute the volume of a cuboid.
# @param width the width of the cuboid
# @return the volume of the cuboid
def volume(width, height, length) :
    return width * height * length
```

Based on the recommendations in the textbook, what change should be made to improve the comments for this function?

1. The `@param` line for width should be removed
2. Additional `@param` lines should be added for height and length
3. The first line should be expanded to describe how the function performs its calculation
4. The `@return` line should be removed

Title

How can the function's comment be improved?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-22

23. You are writing a function that converts from Liters to Gallons. Which function header is the best?

1. `def litersToGallons() :`
2. `def litersToGallons(liters) :`
3. `def litersToGallons(gallons) :`
4. `def litersToGallons(liters, gallons) :`

Title

Which function header is best for a function that converts from Liters to Gallons?

type

mc

Section

5.2 Implementing and Testing Functions

id

testbank-py-1-ch05-23

24. Given the following code snippet, what is considered a parameter variable(s)?

```
def mystery(num1, num2) :
    result = num1 ** num2
    return result
mystery(10, 2)
```

1. 10, 2
2. num1, num2
3. result
4. mystery

Title

Identify the parameter variable(s) in the given code snippet

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-24

25. Given the following code snippet, what is considered an argument(s)?

```
def mystery(num1, num2) :
    result = num1 ** num2
    return result
mystery(10, 2)
```

1. 10, 2
2. num1, num2
3. result
4. mystery

Title

Identify the argument(s) in the given code snippet

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-25

26. Parameter variables should not be changed within the body of a function because

1. This will generate a compiler error
2. This will generate a run-time error
3. It is confusing because it mixes the concept of a parameter with that of a variable
4. It is confusing because parameter variables cannot store values

Title

Why shouldn't parameter variables be changed within the body of a method?

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-26

27. Consider the following program:

```
def squareArea(sideLength) :
    return sideLength ** 2

a = squareArea(4)
```

What are the arguments (actual parameters) in this program?

1. 2
2. 4
3. sideLength
4. squareArea

Title

Identify the arguments in a program

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-27

28. Consider the following program:

```
def main() :
    a = 5
    print(doubleIt(a))

def doubleIt(x) :
    return x * 2

main()
```

What output is generated when this program is run?

1. 2
2. 4
3. 5
4. 10

Title

Trace a main program and function call

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-28

29. Consider the following program:

```
def main() :
    a = 10
    print(doTwice(a))

def doTwice(x) :
    x = x * 2
    x = x * 2
    return x

main()
```

What output is generated when this program is run?

1. 2
2. 10
3. 20
4. 40

Title

Trace a main program and function call

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-29

30. Consider the following program:

```
def main() :
    a = 2
    doubleIt(a)
    print(a)

def doubleIt(x) :
    x = x * 2

main()
```

What output is generated when this program is run?

1. 2
2. 4
3. 8
4. Python reports an error because doubleIt does not contain a return statement

Title

Trace a main program and function call

type

mc

Section

5.3 Parameter Passing

id

testbank-py-1-ch05-30

31. Which statement causes the following function to exit immediately?

```
def mystery(num1, num2) :
    result = num1 ** num2
    return result
mystery(10, 2)
```

1. `mystery(10,2)`
2. `result = num1 ** num2`
3. `return result`
4. None of the statements cause the function to exit immediately.

Title

Which statement causes the function to exit immediately?

type

mc

Section

5.4 Return Values

id

testbank-py-2-ch05-31

from

testbank-py-1-ch05-31

32. Which statement should be added or modified to remove the possibility of a run-time error in this code snippet?

```
1: def floorDivision(value1, value2) :
2:     return value1 // value2
```

1. in line 2: change `//` to `/`
2. in line 2: change `//` to `%`
3. add this statement after line 1: `if value2 == 0 : return 0`
4. add this statement after line 1: `if value2 == 0 : return`

Title

Which statement should be added or modified to remove the run-time error in a code snippet?

type

mc

Section

5.4 Return Values

id

testbank-py-2-ch05-32

from

testbank-py-1-ch05-32

33. What happens in this code snippet if `sideLength = -10`?

```
def cubeSurfaceArea(sideLength) :
    if sideLength >= 0 :
        return 6 * (sideLength * sideLength)
# There are six sides to a cube; surface area of each side is sideLength squared
```

1. the function returns 600
2. the function returns -600
3. an error occurs and aborts the program

4. a special value of `None` will be returned from the function

Title

What is wrong with the following code snippet?

type

mc

Section

5.4 Return Values

id

testbank-py-1-ch05-33

34. How many return statements can be included in a function?

1. Exactly one
2. One or two
3. One, two or more
4. Zero or more

Title

How many return statements can be included in a function?

type

mc

Section

5.4 Return Values

id

testbank-py-2-ch05-34

from

testbank-py-1-ch05-34

35. What is the purpose of this code snippet?

```
def mystery(n) :
    if n % 2 == 0 :
        return True
    else :
        return False
```

1. to determine if `n` is even or odd
2. to find the remainder of `n` divided by 2
3. to find the value of `n` divided by 2
4. to determine if `n` is positive or negative

Title

What is the purpose of this code snippet?

type

mc

Section

5.4 Return Values

id

testbank-py-1-ch05-35

36. When should a computation be turned into a function?

1. when it may not be used
2. when it is only used once

3. when it may be used more than once

4. only if it contains complex mathematically equations

Title

When should a computation be turned into a function?

type

mc

Section

5.4 Return Values

id

testbank-py-1-ch05-36

37. The following program is supposed to display a message indicating if the integer entered by the user is even or odd. What is wrong with the program?

```
num = int(input("Enter an integer: "))
print("The integer is", evenOdd(num))
```

```
def evenOdd(n) :
    if n % 2 == 0 :
        return "even"
    return "odd"
```

1. The function definition must appear before the function is called.

2. The input and print statements must reside in a function named `main`.
3. The variable `num` and the parameter variable `n` must have the same name.
4. An `else` clause must be added to the if statement.

Title

What is wrong with this program for determining if an integer is even or odd?

type

mc

Section

5.4 Return Values

id

testbank-py-1-ch05-37

38. The following function is supposed to return -1 when `x` is negative, +1 when `x` is positive, or 0 if `x` is zero. What, if anything, is wrong with the function?

```
def plusMinusZero(x) :
    if x == 0 :
        return 0
    elif x <= 0 :
        return -1
    else x >= 0 :
        return 1
```

1. A return statement must be added at the end of the function
2. Both occurrences of `elif` must be replaced with `if`
3. The `<=` and `>=` must be replaced with `<` and `>`
4. Nothing is wrong with the function

Title

What is wrong with the function?

type
mc
Section
5.4 Return Values
id
testbank-py-1-ch05-38

39. What is wrong with the following function for computing the amount of tax due on a purchase?

```
def taxDue(amount, taxRate) :
    amount = amount * taxRate

def main() :
    . . .
    total = taxDue(subtotal, TAX_RATE)
    . . .
```

1. The amount of tax due is not computed correctly
2. The function must print a value
3. The function must return a value
4. The function must take an additional parameter

Title
What is wrong with the function?
type
mc
Section
5.4 Return Values
id
testbank-py-1-ch05-39

40. The purpose of a function that does not return a value is

1. to package a repeated task as a function even though the task does not yield a value
2. to insert a temporary implementation of a function that can be refined later
3. to provide a function that can only be included in an assignment statement
4. only used when the function needs to produce output

Title
What is the purpose of methods without a return value?
type
mc
Section
5.5 Functions Without Return Values
id
testbank-py-1-ch05-40

41. Which function call correctly invokes the partial drawShape function listed below and prints a star triangle?

```
def drawShape(type) :
    length = len(type)
    if length == 0 :
        return
    if type == "triangle" :
```

```

    print("  *")
    print(" ***")
    print("*****")
drawShape("triangle")

```

1. drawShape(triangle)
2. drawShape("triangle")
3. drawShape
4. value = drawShape(triangle)

Title

Which call correctly invokes a function?

type

mc

Section

5.5 Functions Without Return Values

id

testbank-py-2-ch05-41

from

testbank-py-1-ch05-41

42. Consider the following functions:

```

def printIt(x) :
    print(x)

def incrementIt(x) :
    return x + 1

def decrementIt(x) :
    return x - 1

def doubleIt(x) :
    return x * 2

```

Which of the following function calls is **not** a reasonable thing to do?

1. print(printIt(5))
2. print(incrementIt(5))
3. print(decrementIt(5))
4. print(doubleIt(5))

Title

Which of the following function calls is unreasonable?

type

mc

Section

5.5 Functions Without Return Values

id

testbank-py-1-ch05-42

43. The following function is supposed to compute and display the value of n-factorial for integers greater than 0.

```

def factorial(n) :
    result = 1

```



```
for i in range(1, n + 1) :
    result = result * i
```

What is wrong with this function?

1. The indenting is wrong. All of the lines should be indented by the same amount.
2. The calculation is wrong. The `result` variable will have something other than n-factorial stored in it.
3. The function is missing a line. A `print` statement must be added at the end of it.
4. The function is missing a line. A `return` statement must be added at the end of it.

Title

What is wrong with the function that is supposed to compute n-factorial?

type

mc

Section

5.5 Functions Without Return Values

id

testbank-py-1-ch05-43

44. The function below randomly generates a number between 1 and 6 to represent a single die. Which implementation listed below allows for other types of die?

```
def die() :
    return randint(1, 6)
```

1.

```
def die(low, high) :
    return
```
2.

```
def die(low, high) :
    return randint(low, high)
```
3.

```
def die(high) :
    return randint(0, high)
```
4.

```
def die(low, high) :
    return high % low
```

Title

Which implementation makes the die function reusable for other types of die?

type

mc

Section

5.6 Problem Solving: Reusable Functions

id

testbank-py-1-ch05-44

45. Given these two separate functions, which implementation combines them into one reusable function?

```
def sixSidedDie() :
    return randint(1, 6)
def fourSidedDie() :
    return randint(1, 4)
```

1.

```
def die(low, high) :  
    return
```
2.

```
def die(low, high) :  
    return high % low
```
3.

```
def die(high) :  
    return randint(0, high)
```
4.

```
def die(low, high) :  
    return randint(low, high)
```

Title

Given these two separate functions, which implementation combines them into one reusable function?

type

mc

Section

5.6 Problem Solving: Reusable Functions

id

testbank-py-1-ch05-45

46. What is stepwise refinement?
1. The process of unit testing
 2. The design of pseudocode for black-box functions
 3. The process of breaking complex problems down into smaller, manageable steps
 4. The use of a temporary implementation of a function that can be improved later

Title

What is stepwise refinement?

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-2-ch05-46

from

testbank-py-1-ch05-46

47. Why is hand-tracing or manually walking through the execution of a function helpful?
1. It enforces the "black-box" concept of function design
 2. It makes unit testing unnecessary
 3. It guarantees that the function will compile without errors
 4. It is an effective way to understand a function's subtle aspects

Title

Why is hand-tracing or manually walking through the execution of a function helpful?

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

```

id
    testbank-py-2-ch05-47
from
    testbank-py-1-ch05-47

```

48. When hand-tracing functions, the values for the parameter variables:

1. Need not be traced because they are never returned
2. Are the same each time the function is invoked
3. May be undetermined or missing when the function executes
4. Are determined by the arguments supplied in the code that invokes the function

```

Title
    What is true about the values for parameter variables when hand-tracing functions?
type
    mc
Section
    5.7 Problem Solving: Stepwise Refinement
id
    testbank-py-2-ch05-48
from
    testbank-py-1-ch05-48

```

49. A stub function is

1. A short function
2. A function that has been unit tested
3. A function that acts as a placeholder and returns a simple value so another function can be tested
4. A function that is broken down into smaller steps through stepwise refinement

```

Title
    What is a stub method?
type
    mc
Section
    5.7 Problem Solving: Stepwise Refinement
id
    testbank-py-1-ch05-49

```

50. Consider the following function:

```

def numberToGrade(x) :
    return "X"

```

This function will eventually be rewritten so that it returns the letter grade associated with x grade points. However, at the moment it is incomplete, and always returns the letter x as a placeholder. In its current form, this function is referred to as a:

1. def
2. param
3. refinement
4. stub

Title
Classify the nature of a function
type
mc
Section
5.7 Problem Solving: Stepwise Refinement
id
testbank-py-1-ch05-50

51. The variable name `perfect` in the function `myFun` in the code snippet below is used as both a parameter variable and a variable in the body of the function. Which statement about this situation is true?

```
def myFun(perfect)
    perfect = 0
    return ((perfect - 1) * (perfect - 1))
```

1. This multiple use of the same variable `perfect` will not compile because the scopes overlap
2. While this is legal and will compile in Python, it is confusing
3. Because the scopes of these variables do not overlap, there is no problem
4. This situation rarely occurs and the compiler always issues a warning

Title
Which statement about variable declarations within functions is true?
type
mc
Section
5.8 Variable Scope
id
testbank-py-2-ch05-51
from
testbank-py-1-ch05-51

52. Consider the following code segment:

```
def main() :
    avg = 0
    total = 0
    for i in range(6) :
        iSquared = i * i
        total = total + iSquared
        avg = total / i
    print(total)
    print(avg)
```

Which of the following answers lists all of the local variables in this code segment?

1. `avg, total, i, iSquared`
2. `i, iSquared`
3. `avg, total`
4. `i`

Title
Which variables are considered local variables?
type

mc
 Section
 5.8 Variable Scope
 id
 testbank-py-2-ch05-52
 from
 testbank-py-1-ch05-52

53. Given the following code snippet, which statement correctly allows the function to update the global variable `total`?

```
1. total = 0
2. def main() :
3.     avg = 0
4.     for i in range(6) :
5.         iSquared = i * i
6.         total = total + iSquared
7.     avg = total / i
8.     print(total)
9.     print(avg)
```

1. add the keyword `global` to line 1
2. line 1 already allows the `total` variable to be updated
3. move line 1 inside the function definition
4. add the statement `global total` after line 2

Title
 How can global variables be updated in a function definition?
 type
 mc
 Section
 5.8 Variable Scope
 id
 testbank-py-1-ch05-53

54. What is the output from the following Python program?

```
def myFun(perfect) :
    perfect = 0
    return ((perfect - 1) * (perfect - 1))

def main() :
    for i in range(4) :
        print(myFun(i), end = " ")

main()
```

1. 1 1 1 1
2. -1 0 1 4
3. 0 0 0 0
4. 1 0 1 4

Title
 What is the output from the code snippet?

type
mc
Section
5.8 Variable Scope
id
testbank-py-1-ch05-54

55. What is the output from the following Python program?

```
def main() :
    a = 10
    r = cubeVolume()
    print(r)

def cubeVolume() :
    return a ** 3

main()
```

1. 10
2. 30
3. 1000
4. Nothing, there is an error.

Title
What is the output from a code snippet involving multiple functions?
type
mc
Section
5.8 Variable Scope
id
testbank-py-2-ch05-55
from
testbank-py-1-ch05-55

56. Which line of code in the Python program below is the recursive invocation of function `myFun`?

```
1: def main() :
2:     for i in range(4) :
3:         print(myFun(i), end = " ")
4: def myFun(perfect) :
5:     perfect = 0
6:     return ((perfect - 1) * (perfect - 1))
7: main()
```

1. Line 1
2. Line 3
3. Line 6
4. There is no recursive invocation in this code segment

Title
Which line of code is the recursive invocation?
type
mc
Section

5.9 Recursive Functions (Optional)

id

testbank-py-2-ch05-56

from

testbank-py-1-ch05-56

57. Which of the following is NOT a good practice when developing a computer program?

1. Put as many statements as possible into the main function
2. Document the purpose of each function parameter
3. Decompose a program into many small functions
4. Place code that is used multiple times into a separate function

Title

Which is NOT a good practice when developing a computer program?

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-2-ch05-57

from

testbank-py-1-ch05-57

58. Which of the following statements about variables is true?

1. A variable is visible from the point at which it is defined until the end of the program.
2. You should use global variables whenever possible..
3. The same name can be used for two different variables in a single method.
4. The same variable name can be used in two different methods.

Title

Which statement about variables is true?

type

mc

Section

5.8 Variable Scope

id

testbank-py-1-ch05-58

59. The `ceil` function in the Python standard library `math` module takes a single value `x` and returns the smallest integer that is greater than or equal to `x`. Which of the following is true about `ceil(56.75)`?

1. The argument is 56.75, and the return value is 57
2. The argument is 56.75, and the return value is 56
3. The argument is 57, and the return value is 56.75
4. The argument is 56, and the return value is 56.75

Title

Which is true about `Math.ceil(56.75)`?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-1-ch05-59

60. Consider a function named `avg`, which accepts four numbers and returns their average. Which of the following is a correct call to the function `avg`?

1. `avg(2, 3.14, 3, 5, 6)`
2. `average = avg(2, 3.14, 4, 5)`
3. `avg()`
4. `average = avg("2", "3", "4", "5")`

Title

Which statement correctly calls the average function with 4 parameters?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-60

from

testbank-py-1-ch05-60

61. Which of the following statements is true about functions in Python?
1. Functions can have only one argument and can return only one return value.
 2. Functions can have multiple arguments and can return multiple return values.
 3. Functions can have multiple arguments and can return one return value.
 4. Functions can have one argument and can return multiple return values.

Title

Which statement is true about functions?

type

mc

Section

5.1 Functions as Black Boxes

id

testbank-py-2-ch05-61

from

testbank-py-1-ch05-61

62. A programmer notices that the following code snippet uses the same algorithm for computing interest earned, but with different variables, in the two places shown below and in several other places in the program. What could be done to improve the program?

```
RATE1 = 10
RATE2 = 5.5
interest = investment * RATE1 / 100
. . .
balance = balance + balance * RATE2 / 100
```

1. Declare the rates as variables, not constants.
2. Define a function that looks up interest rates.

3. Define a function that prompts the user for an amount and a rate of interest, then returns the interest earned.
4. Define a function that computes the interest earned from an amount and a rate of interest.

Title

What could be done to improve (program that does same calculation with different values)?

type

mc

Section

5.6 Problem Solving: Reusable Functions

id

testbank-py-1-ch05-62

63. What is wrong with the following code?

```
def grade(score) :
    if score >= 90 :
        return "A"
    elif score >= 80 :
        return "B"
    elif score >= 70 :
        return "C"
    elif score >= 60 :
        return "D"
```

1. The name of the parameter variable is illegal
2. The type of the parameter variable is invalid
3. Another `return` statement needs to be added to the function
4. One of the existing `return` statements is not correct

Title

What is wrong with a function that includes multiple return statements?

type

mc

Section

5.6 Problem Solving: Reusable Functions

id

testbank-py-2-ch05-63

from

testbank-py-1-ch05-63

64. Given the following code, what is the output?

```
def main() :
    i = 20
    b = mysteriousFunction2(i)
    print(b + i)

def mysteriousFunction1(i) :
    n = 0
    while n * n <= i :
        n += 1
    return n - 1

def mysteriousFunction2(a) :
    b = 0
```

```

    for n in range(a) :
        i = mysteriousFunction1(n)
        b = b + i
    return b

main()

```

1. 50
2. 60
3. 70
4. 80

Title

What is output by a code segment where one function calls another function?

type

mc

Section

5.4 Return Values

id

testbank-py-2-ch05-64

from

testbank-py-1-ch05-64

65. For a program that reads city names repeatedly from the user and calculates the distance from a company's headquarters, which of the following would be a good design based on stepwise refinement?
1. Write on function that calculates distance randomly
 2. Write one function that reads city name
 3. Write one function that reads city name and another function that calculates distance
 4. Write one function that reads distance and finds city name

Title

For a program that reads city names repeatedly from the user and calculates the distance from a company's headquarters, which of the following would be a good design based on stepwise refinement?

type

mc

Section

5.4 Return Values

id

testbank-py-1-ch05-65

66. For a program that reads three letter grades and calculates an average of those grades, which of the following would be a good design based on stepwise refinement?
1. Write one function that reads three letter grades, converts each letter grade to a number, and calculates the average of the three numbers.
 2. Write one function that reads three letter grades, and a second function to convert each letter to a number and calculate the average of the three numbers.
 3. Write one function that reads a letter grade and returns the number equivalent, and one function that computes the average of three numbers.
 4. Stepwise refinement cannot be applied to this problem.

Title

For a program that reads three letter grades and calculates an average of those grades, which of the following would be a good design based on stepwise refinement?

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-1-ch05-66

67. What is the output of the following code snippet?

```
def main() :
    print(blackBox(4))

def blackBox(a) :
    if a <= 0 :
        val = 1
    else :
        val = a + blackBox(a - 2)
    return val
```

main()

1. 4
2. 2
3. 1
4. 7

Title

What is output of snippet (with recursive function)?

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-67

68. What is the output of the following code snippet?

```
def main() :
    print("fun(2) =", fun(2))

def fun(a) :
    returnValue = 0
    if a > 5 :
        returnValue = a
    else :
        returnValue = fun(2 * a)
    return returnValue
```

main()

1. fun(2) = 4
2. fun(2) = 8
3. fun(2) = 16
4. fun(2) = 32

Title
What is output by a code segment that includes recursion?
type
mc
Section
5.9 Recursive Functions (Optional)
id
testbank-py-2-ch05-68
from
testbank-py-1-ch05-68

69. Based on the code snippet, which of the following statements is correct?

```
def main() :
    reoccur(1)

def reoccur(a) :
    print(a)
    reoccur(a + 1)

main()
```

1. The code snippet gives a compilation error as the reoccur function cannot call itself.
2. The code snippet executes and infinitely recurses, displaying 1, 2, 3, 4, and so on.
3. The code snippet executes and displays 1.
4. The code snippet executes and does not produce any output.

Title
What is output of snippet (with recursive function)?
type
mc
Section
5.9 Recursive Functions (Optional)
id
testbank-py-1-ch05-69

70. What is the output if the function call is `testMyVal(6)` in the following code snippet?

```
def testMyVal(a) :
    if a > 0 :
        testMyVal(a - 2)
    print(a, end = " ")
```

1. 0 2 4 6
2. 0 0 0 0
3. The code snippet executes and infinitely recurses, displaying 2, 4, 5, and so on.
4. 6 4 2 0

Title
What is output of snippet (with recursive function)?
type
mc
Section
5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-70

71. Which of the following code snippets returns the factorial of a given number? (Hint: Factorial of 5 = $5! = 1 * 2 * 3 * 4 * 5 = 120$)

1.

```
def factorial(num) :  
    return num * factorial(num - 1)
```
2.

```
def factorial(num) :  
    if(num == 1) :  
        return 1  
    else :  
        return num * factorial(num)
```
3.

```
def factorial(num) :  
    if(num == 1) :  
        return 1  
    else :  
        print(num * factorial(num - 1))
```
4.

```
def factorial(num) :  
    if(num == 1) :  
        return 1  
    else :  
        return num * factorial(num - 1)
```

Title

Which snippet calculates the factorial of a given number?

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-71

72. Consider this recursive function:

```
def mystery(x) :  
    if x <= 0 :  
        return 0  
    else :  
        return x + mystery(x - 1)
```

What is mystery(5)?

1. 15
2. 10
3. 0
4. 4

Title

What is output of snippet (with recursive function)?

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-72

73. In the code snippet below, which variables are considered global variables:

```
a = 0
b = 5
def main() :
    global a, b
    fun1()
    fun2()

def fun1() :
    i = 0
    b = 0

def fun2() :
    a = b + 1

main()
```

1. a, b, i
2. fun1, fun2
3. a, b
4. i

Title

In the given code snippet, identify the global variables

type

mc

Section

5.8 Variable Scope

id

testbank-py-1-ch05-73

74. Consider the following program:

```
def main() :
    print(factorial(n))          # Line 1

def factorial(n) :              # Line 2
    result = 1
    for i in range(1, n + 1) :
        result = result * i     # Line 3
    return i

main()
```

Which of the following lines is a recursive function call?

1. Line 1
2. Line 2
3. Line 3
4. There are no recursive function calls in the program

Title

Which line contains a recursive function call?

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-74

75. Consider the following program:

```
def main() :
    print(factorial(n))          # Line 1

def factorial(n) :              # Line 2
    if n <= 1 :
        return 1
    return n * factorial(n - 1)  # Line 3

main()
```

What line is the recursive call on?

1. Line 1
2. Line 2
3. Line 3
4. There are no recursive function calls in the program

Title

Which line contains a recursive function call?

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-75

76. What output is generated when the following program runs?

```
def main() :
    x = mystery(5)
    print(x)

def mystery(n) :
    if n == 1 :
        return 1
    return 2 * mystery(n - 1)

main()
```

1. 4
2. 8
3. 16
4. 32

Title

Trace a recursive function

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-76

77. What output is generated when the following program runs?

```
def main() :
    x = mystery(9, 12)
    print(x)

def mystery(a, b) :
    if b == 0 :
        return a
    else :
        return mystery(b, a % b)

main()
```

1. 3
2. 6
3. 9
4. 12

Title

Trace a recursive function

type

mc

Section

5.9 Recursive Functions (Optional)

id

testbank-py-1-ch05-77

78. The tool that allows you to follow the execution of a program and helps you locate errors is known as a(n):

1. Compiler
2. Debugger
3. Interpreter
4. Virtual machine

Title

The tool that allows you to follow the execution of a program and helps you locate errors is known as a(n):

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-2-ch05-78

79. The location where the debugger stops executing your program so that you can inspect the values of variables is called a(n):

1. breakpoint
2. function call
3. inspection point

4. step point

Title

The location where the debugger stops executing your program so that you can inspect the values of variables is called a(n):

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-2-ch05-79

80. Which debugging command allows you to quickly run an entire function instead of examining its body a line at a time?
1. Step into
 2. Step over
 3. Step point
 4. Step quickly

Title

Which debugging command allows you to quickly run an entire function instead of examining its body a line at a time?

type

mc

Section

5.7 Problem Solving: Stepwise Refinement

id

testbank-py-2-ch05-80

81. What term is used to refer to a collection of functions and classes organized into one or more user-defined modules?
1. function argument
 2. hardware toolbox
 3. software toolkit
 4. standard library

Title

What term is used to refer to a collection of functions and classes organized into one or more user-defined modules?

type

mc

Section

5.9 Graphics: Building an Image Processing Toolkit

id

testbank-py-2-ch05-81

82. Which function should **not** be included in a software toolkit for managing student enrollment in courses?
1. computeNegativeImage
 2. enrollStudentInCourse
 3. getGradePointAverage

4. setFinalExamMark

Title

Which function should not be included in a software toolkit for managing student enrollment?

type

mc

Section

5.9 Graphics: Building an Image Processing Toolkit

id

testbank-py-2-ch05-82

83. Consider the following code segment:

```
def f1():
    print("A", end="")

def f2():
    f1()
    print("B", end="")
```

What output is generated when it runs?

1. A
2. AB
3. B
4. The code segment does not display any output.

Title

What is output by the code segment involving functions?

type

mc

Section

5.5 Functions Without Return Values

id

testbank-py-2-ch05-83

84. Consider the following code segment:

```
def f1():
    print("a", end="")
    return "b"

def f2():
    print("c", end="")
    d = f1()
    print(d, end="")
    print("e", end="")

def f3():
    print("f", end="")
    f2()
    print("g", end="")

f3()
```

What output is generated when it runs?

1. fg
2. fceg
3. fcabeg
4. fcadeg

Title

What is output by the code segment involving multiple functions?

type

mc

Section

5.5 Functions Without Return Values

id

testbank-py-2-ch05-84

85. What term is used to describe the portion of a program in which a variable can be accessed?

1. locale
2. region
3. scope
4. volume

Title

What term is used to describe the portion of a program in which a variable can be accessed?

type

mc

Section

5.8 Variable Scope

id

testbank-py-2-ch05-85