

## Homework 5

### Functions

---

#### TRUE/FALSE

1. Python function names follow the same rules as those for naming variables.

ANS:

2. The function header marks the beginning of the function definition.

ANS:

3. A function definition specifies what a function does and causes the function to execute.

ANS:

4. A hierarchy chart shows all the steps that are taken inside a function.

ANS:

5. The pass keyword is used to pass arguments to a function.

ANS:

6. A local variable can be accessed from anywhere in the program.

ANS:

7. Different functions can have local variables with the same names.

ANS:

8. Python allows you to pass multiple arguments to a function.

ANS:

9. To assign a value to a global variable in a function, the global variable must be first declared in the function.

ANS:

10. The value assigned to a global constant can be changed in the mainline logic.

ANS:

11. One reason not to use global variables is that it makes a program hard to debug.

ANS:

12. A value-returning function is like a simple function except that when it finishes it returns a value back to the part of the program that called it.

ANS:

13. Unlike other languages, in Python the number of values a function can return is limited to one.

ANS:

14. In Python you can have a list of variables on the left side of the argument operator.

ANS:

15. In Python there is no restriction on the name of a module file.

ANS:

16. One of the drawbacks of a modularized program is that the only structure you can use in such a program is the sequence structure.

ANS:

17. One reason to store graphics functions in a module is so that you can import the module into any program that needs to use those functions.

ANS:

18. The **randrange** function returns a randomly selected value from a specific sequence of numbers.

ANS:

19. The **math** function **atan(x)** returns one tangent of **x** in radians.

ANS:

20. The **math** function **ceil(x)** returns the smallest integer that is greater than or equal to **x**.

ANS:

21. Unfortunately, there is no way to store and call on functions when using turtle graphics.

ANS:

## MULTIPLE CHOICE

1. What is a group of statements that exists within a program for the purpose of performing a specific task?
  - a. a function
  - b. a subtask
  - c. a process
  - d. a subprocess

ANS:

2. The first line in a function definition is known as the function
  - a. header
  - b. block
  - c. return
  - d. parameter

ANS:

3. The \_\_\_\_\_ design technique can be used to break down an algorithm into functions.
  - a. subtask
  - b. block
  - c. top-down
  - d. simplification

ANS:

4. A set of statements that belong together as a group and contribute to the function definition is known as a
  - a. header
  - b. block
  - c. return
  - d. parameter

ANS:

5. A(n) \_\_\_\_\_ chart is also known as a structured chart.
  - a. flow
  - b. data
  - c. hierarchy
  - d. organizational

ANS:

6. The \_\_\_\_\_ keyword is ignored by the Python interpreter and can be used as a placeholder for code that will be written later.
  - a. **placeholder**
  - b. **pass**
  - c. **pause**
  - d. **skip**

ANS:

7. A \_\_\_\_\_ variable is created inside a function.
- global
  - constant
  - named constant
  - local

ANS:

8. The \_\_\_\_\_ of a local variable is the function in which that variable is created.
- global reach
  - definition
  - space
  - scope

ANS:

9. A(n) \_\_\_\_\_ is any piece of data that is passed into a function when the function is called.
- global variable
  - argument
  - local variable
  - parameter

ANS:

10. A(n) \_\_\_\_\_ is a variable that receives an argument that is passed into a function.
- global variable
  - argument
  - named constant
  - parameter

ANS:

11. A \_\_\_\_\_ variable is accessible to all the functions in a program file.
- keyword
  - local
  - global
  - string

ANS:

12. A \_\_\_\_\_ constant is a name that references a value that cannot be changed while the program runs.
- keyword
  - local
  - global
  - string

ANS:

13. When a function is called by its name during the execution of a program, then it is

- a. executed
- b. located
- c. defined
- d. exported

ANS:

14. It is recommended that programmers avoid using \_\_\_\_\_ variables in a program whenever possible.
- a. local
  - b. global
  - c. string
  - d. keyword

ANS:

15. The Python library functions that are built into the Python \_\_\_\_\_ can be used by simply calling the required function.
- a. code
  - b. compiler
  - c. linker
  - d. interpreter

ANS:

16. Python comes with \_\_\_\_\_ functions that have already been prewritten for the programmer.
- a. standard
  - b. library
  - c. custom
  - d. key

ANS:

17. What type of function can be used to determine whether a number is even or odd?
- a. even
  - b. odd
  - c. math
  - d. Boolean

ANS:

18. A value-returning function is
- a. a single statement that performs a specific task
  - b. called when you want the function to stop
  - c. a function that will return a value back to the part of the program that called it
  - d. a function that receives a value when called

ANS:

19. Which of the following statements causes the interpreter to load the contents of the **random** module into memory?
- a. **load random**
  - b. **import random**

- c. **upload random**
- d. **download random**

ANS:

20. Which of the following will assign a random integer in the range of **1** through **50** to the variable **number**?
- a. **random(1, 50) = number**
  - b. **number = random.randint(1, 50)**
  - c. **randint(1, 50) = number**
  - d. **number = random(range(1, 50))**

ANS:

21. What does the following statement mean?

```
num1, num2 = get_num()
```

- a. The function **get\_num()** is expected to return a value for **num1** and for **num2**.
- b. The function **get\_num()** is expected to return one value and assign it to **num1** and **num2**.
- c. This statement will cause a syntax error.
- d. The function **get\_num()** will receive the values stored in **num1** and **num2**.

ANS:

22. What will display after the following code is executed?

```
def main():  
    print("The answer is", magic(5))
```

```
def magic(num):  
    answer = num + 2 * 10  
    return answer
```

```
if __name__ == '__main__':  
    main()
```

- a. **70**
- b. **25**
- c. **100**
- d. The statement will cause a syntax error.

ANS:

23. In a value-returning function, the value of the expression that follows the keyword \_\_\_\_\_ will be sent back to the part of the program that called the function.
- a. **def**
  - b. **result**
  - c. **sent**
  - d. **return**

ANS:

24. The Python standard library's \_\_\_\_\_ module contains numerous functions that can be used in mathematical calculations.

- a. **math**
- b. **string**
- c. **random**
- d. **number**

ANS:

25. Which of the following functions returns the largest integer that is less than or equal to its argument?

- a. **floor**
- b. **ceil**
- c. **lesser**
- d. **greater**

ANS:

26. What will be the output after the following code is executed?

```
def pass_it(x, y):  
    z = x + ", " + y  
    return(z)  
  
name2 = "Julian"  
name1 = "Smith"  
fullname = pass_it(name1, name2)  
print(fullname)
```

- a. **Julian Smith**
- b. **Smith Julian**
- c. **Julian, Smith**
- d. **Smith, Julian**

ANS:

27. What will be the output after the following code is executed?

```
def pass_it(x, y):  
    z = x , ", " , y  
  
num1 = 4  
num2 = 8  
answer = pass_it(num1, num2)  
print(answer)
```

- a. **4, 8**
- b. **8, 4**
- c. **48**
- d. **None**

ANS:

28. What will be the output after the following code is executed?

```
def pass_it(x, y):  
    z = y**x  
    return(z)  
  
num1 = 3
```

```
num2 = 4
answer = pass_it(num1, num2)
print(answer)
```

- a. 81
- b. 64
- c. 12
- d. None

ANS:

29. What will be displayed after the following code is executed?

```
def pass_it(x, y):
    z = x*y
    result = get_result(z)
    return(result)

def get_result(number):
    z = number + 2
    return(z)

num1 = 3
num2 = 4
answer = pass_it(num1, num2)
print(answer)
```

- a. 12
- b. 9
- c. 14
- d. Nothing, this code contains a syntax error.

ANS:

30. What does the following program do?

```
import turtle
def main():
    turtle.hideturtle()
    square(100,0,50,'blue')

def square(x, y, width, color):
    turtle.penup()
    turtle.goto(x, y)
    turtle.fillcolor(color)
    turtle.pendown()
    turtle.begin_fill()
    for count in range(4):
        turtle.forward(width)
        turtle.left(90)
    turtle.end_fill()

if __name__ == '__main__':
    main()
```

- a. It draws a blue square at coordinates (100, 0), 50 pixels wide, starting at the top right corner.
- b. It draws a blue square at coordinates (0, 50), 100 pixels wide, starting at the top right corner.
- c. It draws a blue square at coordinates (100, 0), 50 pixels wide, in the lower-left corner.
- d. Nothing since you cannot call a function with turtle graphics.

ANS:

31. What does the following program do?

```
import turtle
def main():
    turtle.hideturtle()
    square(100,0,50,'blue')

def square(x, y, width, color):
    turtle.penup()
    turtle.goto(x, y)
    turtle.fillcolor(color)
    turtle.pendown()
    turtle.begin_fill()
    for count in range(2):
        turtle.forward(width)
        turtle.left(90)
    turtle.end_fill()

if __name__ == '__main__':
    main()
```

- a. It draws a blue square.
- b. It draws a blue triangle.
- c. It draws 2 blue lines.
- d. Nothing since you cannot call a function with turtle graphics.

ANS:

## COMPLETION

1. The code for a function is known as a function \_\_\_\_\_.

ANS:

2. The function header begins with the keyword \_\_\_\_\_ and is followed by the name of the function.

ANS:

3. The **main** function contains a program's \_\_\_\_\_ logic which is the overall logic of the program.

ANS:

4. In a flowchart, a function call is depicted by a(n) \_\_\_\_\_.

ANS:

5. The top-down design breaks down the overall task of a program into a series of \_\_\_\_\_.

ANS:

6. A(n) \_\_\_\_\_ chart is a visual representation of the relationships between functions.

ANS:

7. Arguments are passed by \_\_\_\_\_ to the corresponding parameter variables in a function.

ANS:

8. A variable is available only to statements in the variable's \_\_\_\_\_.

ANS:

9. Functions that are in the standard library are stored in files that are known as \_\_\_\_\_.

ANS:

10. To refer to a function in a module, Python uses \_\_\_\_\_ notation.

ANS:

11. A value-returning function has a(n) \_\_\_\_\_ statement that sends a value back to the part of the program that called it.

ANS:

12. The \_\_\_\_\_ chart is an effective tool used by programmers to design and document functions.

ANS:

13. The 'P' in the acronym IPO refers to \_\_\_\_\_.

ANS:

14. In Python, a module's file name should end in \_\_\_\_\_.

ANS:

15. The approach known as \_\_\_\_\_ makes a program easier to understand, test, and maintain.

ANS:

16. The return values of the trigonometric functions in Python are in \_\_\_\_\_.

ANS: