# Functions

Standard difficulty · 20 questions · 40 marks · Time: 43 minutes

Name: _____ Date: _____ Score: ____ / 40

Show your working for Sections B and C. Answers are on the last page — try every question first!

## Section A — Choose the correct answer — 8 × 1 = 8 marks

**1.** The values written in a function call are called:

**(a)** return values **(b)** local variables
**(c)** parameters **(d)** arguments

**2.** What does this print?

```
def area(l, b=5):
    return l * b + 9

print(area(8), area(8, 9))
```

**(a)** 49 54 **(b)** 17 81 **(c)** 50 81 **(d)** 49 81

**3.** What does this print?

```
def total(p, q=2, r=5):
    return p + q * r

print(total(1, 9))
```

**(a)** 46 **(b)** 15 **(c)** 11 **(d)** 64

**4.** What does this print?

```
x = 13
def change():
    global x
    x = x + 8

change()
change()
print(x)
```

**(a)** 37 **(b)** 13 **(c)** 29 **(d)** 21

**5.** What does this print?

```
x = 11
def show():
    x = 45
    return x * 3

print(show(), x)
```

**(a)** 135 11 **(b)** 33 11 **(c)** 33 45 **(d)** 135 45

**6.** What does this print?

```
def calc(a, b):
    return a + b, a * b

s, p = calc(3, 8)
print(p, s)
```

**(a)** 24 11 **(b)** 24 24 **(c)** (24, 11) **(d)** 25 11

**7.** Which call of this function gives an error?

```
def show(name, age, city='Delhi'):
    print(name, age, city)
```

**(a)** show(age=16, name='Zoya') **(b)** show('Zoya', 16)

**(c)** show(city='Pune') **(d)** show('Zoya', city='Agra', age=16)

**8.** What does this print?

```
import math
print(math.pow(6, 3), 6 ** 3)
```

**(a)** 217.0 216 **(b)** 215.0 216 **(c)** 216.0 216 **(d)** 18.0 18

## Section B — Short answer — 8 × 2 = 16 marks

**9.** What is the difference between a parameter and an argument?

**10.** What does this print? Give (a) the first value (b) the second.

```
def area(l, b=9):
    return l * b + 9

print(area(6), area(6, 8))
```

**11.** What does this print?

```
def total(p, q=4, r=5):
    return p + q * r

print(total(7, r=5))
```

**12.** What does this print?

```
x = 10
def change():
    global x
    x = x * 2

change()
change()
change()
print(x)
```

**13.** What does this print?

```
def update(L, n):
    for i in range(len(L)):
        if L[i] % 2 == 0:
            L[i] = L[i] + 5
    n = n + 5

nums = [27, 18, 17, 13]
m = 10
update(nums, m)
print(nums, m)
```

**14.** Write the letters this program prints, in order, separated by commas.

```
def one():
    print('S')

def two():
    print('T')
    one()

two()
two()
two()
two()
```

**15.** What does this program print? Give (a) the first line (b) the second.

```
def double(n):
    print(n * 5)

r = double(12)
print(r)
```

**16.** What does this print?

```
def power(x, n=4):
    return x ** n

print(power(3) + power(9, 1))
```

## Section C — Word problems — 4 × 4 = 16 marks

**17.** Explain with an example how default and keyword arguments work, and the rules for using them.

**18.** Write the output of each program and explain the difference.

(a)

```
def update(L, n):
    for i in range(len(L)):
        if L[i] % 2 == 0:
            L[i] = L[i] + 9
    n = n + 9

nums = [24, 27, 18, 10]
m = 7
update(nums, m)
print(nums, m)
```

(b)

```
x = 15
def show():
    x = 57
    return x * 5

print(show(), x)
```

**19.** Trace this program. Give (a) the first line it prints (b) the second (c) the third.

```
def stats(L, limit=21):
    big = []
    for x in L:
        if x > limit:
            big.append(x)
    return len(big), sum(big)

n, s = stats([9, 46, 22, 36, 28, 11])
print(n)
print(s)
print(stats([9, 46, 22, 36, 28, 11], 31))
```

**20.** (a) What does this program print? (b) How many times is add() called? (c) Why is the word global needed inside add()?

```
count = 0

def add(x):
    global count
    count = count + 1
    return x

total = 0
x = 2
for i in range(6):
    total = total + add(x)
    x = x + 3
print(total, count)
```

**Want another sheet on this chapter?** In the Ambiful Worksheets app every sheet is new, with its answer key, and your child never gets the same question twice.

Scan the code to get the free app, or open **worksheets.ambiful.com/get**

# Answer key

Functions · AW-12CS-FNC-2026-V1 · For parents — fold back or tear off before handing the sheet to your child.

| Q | Answer | Working |
|---|---|---|
| 1 | **(d) arguments** | — |
| 2 | **(d) 49 81** | b takes its default value only when it is not passed |
| 3 | **(a) 46** | match keyword arguments by name, the rest by position; missing ones take their defaults |
| 4 | **(c) 29** | global x makes the function change the variable outside it |
| 5 | **(a) 135 11** | the x inside show() is a new local variable; the global x is unchanged |
| 6 | **(a) 24 11** | return a + b, a * b returns a tuple of two values |
| 7 | **(c) show(city='Pune')** | name and age are missing |
| 8 | **(c) 216.0 216** | math.pow() always returns a float; ** on two ints gives an int |
| 9 | **A parameter is the name in the function header; an argument is the value passed in the function call.** | — |
| 10 | **(a) 63; (b) 57** | b takes its default value only when it is not passed |
| 11 | **27** | match keyword arguments by name, the rest by position; missing ones take their defaults |
| 12 | **80** | global x makes the function change the variable outside it |
| 13 | **[27, 23, 17, 13] 10** | the list is the same object inside the function, so changes to it stay; n is a new local name |
| 14 | **T, S, T, S, T, S, T, S** | a def only defines a function; it runs when called, and two() calls one() |
| 15 | **(a) 60; (b) None** | double() prints but returns nothing, so r is None |
| 16 | **90** | 3 ** 4 = 81, then + 9 |
| 17 | **def f(a, b=2, c=3): return a + b * c. f(1) = 7, f(1, 4) = 13, f(1, c=5) = 11, f(c=5, a=1) = 11. Default parameters must come after the others in the header; in a call, keyword arguments come after positional ones and no parameter may get two values.** | — |
| 18 | **(a) [33, 27, 27, 19] 7; (b) 285 15. A list passed to a function is the same object, so changing its items is seen outside; assigning to a name inside a function makes a new local variable.** | mutable list changed in place; local names hide global ones |
| 19 | **(a) 4; (b) 132; (c) (2, 82)** | above 21: 46, 22, 36, 28; above 31: 46, 36 |
| 20 | **(a) 57 6; (b) 6; (c) without it, count = count + 1 makes count local and reading it first gives an UnboundLocalError.** | x takes 2, 5, 8, 11, 14, 17 |