Assessment – Answers

# Sequence

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2 | location = "Leeds"  print("I live in", location) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. "I live in", location
  2. "I live in", "Leeds"
  3. I live in location
  4. **I live in Leeds**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | print("Where do you live?")  location = input()  print("I’ve never been to", location) |

When this program is executed, what will be displayed on the screen, as a result of executing **line 3**?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **I’ve never been to and whatever the user has typed at the keyboard**
  2. I’ve never been to location
  3. I’ve never been to input()
  4. It is not possible to know the output without executing the program.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2 | answer = 3 + 13 \* 3  print("The Answer is", answer) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. The Answer is 3 + 13 \* 3
  2. **The Answer is 42**
  3. The Answer is 48
  4. The Answer is answer

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | b = 42  a = 13  print(a,b) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **13 42**
  2. 42 13
  3. a b
  4. There is an error in the program because variables a and b are not assigned values in the right order.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | print(a,b)  b = 42  a = 13 |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 13 42
  2. 42 13
  3. There is an error in the program because variables a and b are not assigned values in the right order.
  4. **There is an error in the program because when line 1 is executed, variables a and b have not been assigned values.**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | a = 13  a = 42  print(a) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 13
  2. **42**
  3. 13 42
  4. There is an error in the program because variable a cannot hold two values at the same time.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | a = 13  b = 42  a = b  print(a, b) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 13 42
  2. 13 13
  3. 42 13
  4. **42 42**
  5. There is an error in the program because when line 4 is executed, variable b no longer has a value.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | a = 13  b = a + 1  print(a, b) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **13 14**
  2. 13 a+1
  3. a b
  4. There is an error in the program because when line 3 is executed, variable a no longer has a value.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | a = 13  a = a + 1  print(a) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 13
  2. **14**
  3. 13 14
  4. There is an error in the program because there are no values for variable a that could be used in the equation in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | a = 13  b = a + 1  a = 42  print(b) |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **14**
  2. 43
  3. 14 43
  4. There is an error in the program because variable b cannot hold two values at the same time.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | a = 13  print(2\*a)  print(a) |

When this program is executed, what will be displayed on the screen, as a result of executing **line 3**?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **13**
  2. 26
  3. 13 26
  4. Line 3 will not be executed because there is an error in line 2: print cannot display expressions such as 2\*a.

# Selection

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("What’s your favourite programming language?")  language = input()  if language != "Python":  print("You should try a little Python too")  else:  print("Hello Pythonista") |

When this program is executed, what will be displayed on the screen, if the user enters Python at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. You should try a little Python too
  2. **Hello Pythonista**
  3. You should try a little Python too

Hello Pythonista

* 1. Hello Pythonista

You should try a little Python too

* 1. There is an error in the program, because there should not be any quotes around "Python" in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("What’s your favourite programming language?")  language = input()  if language == "Python":  print("You should try a little Python too")  else:  print("Hello Pythonista") |

When this program is executed, what will be displayed on the screen, if the user enters Python at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **You should try a little Python too**
  2. Hello Pythonista
  3. You should try a little Python too

Hello Pythonista

* 1. Hello Pythonista

You should try a little Python too

* 1. There is an error in the program, because there should not be any quotes around "Python" in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("What’s your favourite programming language?")  language = input()  if language != Python:  print("You should try a little Python too")  else:  print("Hello Pythonista") |

When this program is executed, what will be displayed on the screen, if the user enters Python at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. You should try a little Python too
  2. Hello Pythonista
  3. You should try a little Python too

Hello Pythonista

* 1. **There is an error in the program, because there should be quotes around Python in line 3.**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5 | print("What’s your favourite programming language?")  language = input()  if language != "Python":  print("You should try a little Python too")  print("Hello Pythonista") |

When this program is executed, what will be displayed on the screen, if the user enters Python at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. You should try a little Python too
  2. Hello Pythonista
  3. You should try a little Python too

Hello Pythonista

* 1. **Nothing will be displayed on the screen**
  2. There is an error in the program, because there is no else.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("Enter a number")  number = int(input())  if number > 0:  print(number, "is positive")  else:  print(number, "is negative") |

When this program is executed, what will be displayed on the screen if the user enters 0 at the prompt?

* 1. 0 is positive

0 is negative

* 1. 0 is positive
  2. **0 is negative**
  3. The program will not display anything because 0 is neither positive nor negative.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | print("Enter a number")  number = int(input())  if number = 0:  print(number, "is zero")  else:  print(number, "is non-zero") |

When this program is executed, what will be displayed on the screen, if the user enters 0 at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 0 is zero

0 is non-zero

* 1. 0 is zero
  2. 0 is non-zero
  3. **There is a syntax error in the condition in line 3.**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | print("How many minutes?")  mins = int(input())  if mins <= 60:  print(mins, "minutes is less than an hour")  else:  hrs = mins / 60  print(mins, "minutes is", hrs, "hours") |

When this program is executed, what will be displayed on the screen, if the user enters 30 at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 30 minutes is 0.5 hours
  2. **30 minutes is less than an hour**
  3. 30 minutes is less than an hour

30 minutes is 0.5 hours

* 1. There is a syntax error in the condition in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | print("How many minutes?")  mins = int(input())  if mins <= 60:  print(mins, "minutes is less than an hour")  else:  hrs = mins / 60  print(mins, "minutes is", hrs, "hours") |

When this program is executed, what will be displayed on the screen, if the user enters 90 at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **90 minutes is 1.5 hours**
  2. 90 minutes is less than an hour
  3. 90 minutes is less than an hour

90 minutes is 1.5 hours

* 1. There is a syntax error in the condition in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | print("How many minutes?")  mins = int(input())  if mins > 60:  print(mins, "minutes is less than an hour")  else:  hrs = mins / 60  print(mins, "minutes is", hrs, "hours") |

When this program is executed, what will be displayed on the screen, if the user enters 90 at the prompt?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. 90 minutes is 1.5 hours
  2. **90 minutes is less than an hour**
  3. 90 minutes is less than an hour

90 minutes is 1.5 hours

* 1. There is a syntax error in the condition in line 3.

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | number = 13  if number == 0:  print("zero")  number = 0 |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **Nothing will be displayed on the screen.**
  2. zero
  3. 13

0

* 1. 13

0

zero

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | number = 13  if number == 0:  print("zero")  else:  print(number)  number = 0 |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. Nothing will be displayed on the screen.
  2. **13**
  3. 13

zero

* 1. 13

zero

0

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | if number == 0:  print("zero")  else:  print(number)  number = 13  number = 0 |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **There is an error in the program because when line 1 is executed, variable number has not been assigned a value.**
  2. Nothing will be displayed on the screen.
  3. 13

zero

* 1. 13

zero

0

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | number = 13  if number < 10:  print("small")  elif number < 100:  print("medium")  elif number < 1000:  print("large") |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. small
  2. **medium**
  3. medium

large

* 1. large

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | number = 13  if number < 10:  print("small")  if number < 100:  print("medium")  if number < 1000:  print("large") |

When this program is executed, what will be displayed on the screen?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. small
  2. medium
  3. **medium**

**large**

* 1. large

# 

# Iteration

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | count = 3  while count > 0:  print(count)  count = count-1 |

How many times will line 3 be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. None (the condition in line 2 will be False the first time it is checked)
  2. 1
  3. **3**
  4. 4
  5. Infinitely (the condition in line 2 will never become False)

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3 | count = 3  while count > 0:  print(count) |

How many times will line 3 be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. None (the condition in line 2 will be False the first time it is checked)
  2. 1
  3. 3
  4. 4
  5. **Infinitely (the condition in line 2 will never become False)**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | count = 3  while count >= 0:  print(count)  count = count-1 |

How many times will line 3 be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. None (the condition in line 2 will be False the first time it is checked)
  2. 1
  3. 3
  4. **4**
  5. Infinitely (the condition in line 2 will never become False)

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4 | count = 3  while count >= 0:  print(count)  count = count-2 |

How many times will line 3 be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. None (the condition in line 2 will be False the first time it is checked)
  2. 1
  3. **2**
  4. 3
  5. Infinitely (the condition in line 2 will never become False)

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5 | from random import randint  number = randint(1,10)  while number != 10:  print(number)  number = randint(1,10) |

How many times will line 4 be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. None (the condition in line 2 will be False the first time it is checked)
  2. 1
  3. 10
  4. **It is impossible to determine in advance**
  5. Infinitely (the condition in line 2 will never become False)

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5 | from random import randint  number = randint(1,10)  while number != 10:  number = randint(1,10)  print(number) |

When this program is executed, what will be displayed on the screen, as a result of executing **line 5**?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. **10**
  2. 1
  3. It is impossible to determine in advance
  4. There is an error in the program, because line 5 should have been indented

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | pin = "1357"  correct = False  while correct == False:  print("Enter pin")  user = input()  if user == pin:  correct = True  print("Welcome") |

When this program is executed, how many times will **line 4** be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. Line 4 will be executed once.
  2. **Line 4 will be executed at least once.**
  3. Line 4 will be executed 1357 times.
  4. Line 4 will be executed an infinite number of times (the while loop will never terminate).

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | pin = "1357"  correct = False  while correct == False:  print("Enter pin")  user = input()  if user == pin:  print("Welcome") |

When this program is executed, how many times will **line 4** be executed?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. Line 4 will be executed once.
  2. Line 4 will be executed at least once.
  3. Line 4 will be executed 1357 times.
  4. **Line 4 will be executed an infinite number of times (the while loop will never terminate).**

1. Read the Python program below:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | pin = "1357"  print("Enter pin")  user = input()  while user != pin:  print("Enter pin")  user = input()  print("Wrong pin, try again")  print("Welcome") |

When this program is executed, what will be displayed on the screen after **line 6** is executed and the user types 1357 on the keyboard?

**Note:** There may be errors in the program and/or it may not behave as expected.

* 1. Wrong pin, try again
  2. **Wrong pin, try again**

**Welcome**

* 1. Welcome
  2. Wrong pin, try again

Enter pin

1. A set of precise instructions that is meant to solve a problem is .
   1. a computer
   2. **an algorithm**
   3. a program
   4. an interpreter
2. The instructions in an algorithm .
   1. can be expressed in any language
   2. **can be expressed in any language, as long as they are precise**
   3. can only be expressed using a programming language
   4. can only be expressed using binary digits
3. The instructions in an algorithm, when expressed precisely in English,

.

* 1. **can only be carried out by humans**
  2. can only be carried out by computers
  3. can be carried out by both humans and computers

1. The instructions in a program, when expressed in Python, .
   1. can only be carried out by humans, as long as they understand Python
   2. can only be carried out by computers
   3. can be carried out by computers, as long as an interpreter is available
   4. **can be carried out by humans, as long as they understand Python, and computers, as long as an interpreter is available**
2. The instructions in a program .
   1. can be expressed in any language
   2. can be expressed in any language, as long as they are precise
   3. **can only be expressed using a programming language**
   4. can only be expressed using binary digits
3. A Python program requires to be executed.
   1. a computer
   2. a program called ‘the Python translator’
   3. **a program called ‘the Python interpreter’**
   4. a program called ‘a development environment’

Resources are updated regularly — the latest version is available at: [ncce.io/tcc](http://ncce.io/tcc).

This resource is licensed under the Open Government Licence, version 3. For more information on this licence, see [ncce.io/ogl](about:blank).