# [CC03] quiz 1 - Python Programming Code

Question **1**

Correct

Marked out of 1.00

Flag question

Question text

Write a Python function, whose name is area, which accepts the radius of a circle as an input parameter and return the area

**For example:**

| **Test** | **Result** |
| --- | --- |
| res = area(1.1)  expect = 3.8013271108436504  delta = 0.000000001  print((res > expect - delta) and (res < expect + delta)) | True |

Answer:(penalty regime: 0 %)

1

2

import math

def area(r): return math.pi \* r\*\*2

CheckQuestion 1

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | res = area(2.6)  expect = 21.237166338267002  delta = 0.000000001  print((res > expect - delta) and (res < expect + delta)) | True | True |  |
|  | res = area(12.6)  expect = 498.75924968391547  delta = 0.000000001  print((res > expect - delta) and (res < expect + delta)) | True | True |  |
|  | res = area(102.45)  expect = 32974.164346060104  delta = 0.000000001  print((res > expect - delta) and (res < expect + delta)) | True | True |  |

Passed all tests!

Top of Form

### Question 2

Correct

Marked out of 1.00

Flag question

#### Question text

Write a Python function **check(lst,n)** to test whether all numbers of the list lst is greater than the number n.

**For example:**

| **Test** | **Result** |
| --- | --- |
| print(check([21,12,5,8],3)) | True |

Answer:(penalty regime: 0 %)

1

2

3

4

import math

def check(lst, n):

return all(num > n for num in lst )

#### CheckQuestion 2

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | print(check([21,12,5,8],3)) | True | True |  |
|  | print(check([21,12,5,8],7)) | False | False |  |
|  | print(check([21,12,5,8],22)) | False | False |  |
|  | print(check([21,12,1000,100,90],11)) | True | True |  |
|  | print(check([21,12,1000,100,90],12)) | False | False |  |

Passed all tests!

Bottom of Form

### Question 3

Correct

Marked out of 1.00

Flag question

#### Question text

Write a Python function gcd to return the greatest common divisor (GCD) of two positive integer parameters

**For example:**

| **Test** | **Result** |
| --- | --- |
| print(gcd(24,36)) | 12 |

Answer:(penalty regime: 0 %)

1

2

3

import math

def gcd(a: int, b: int) -> int:

return math.gcd(a, b)

CheckQuestion 3

#### Feedback

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | print(gcd(24,36)) | 12 | 12 |  |
|  | print(gcd(24,9)) | 3 | 3 |  |
|  | print(gcd(54,36)) | 18 | 18 |  |
|  | print(gcd(1071,462)) | 21 | 21 |  |

Passed all tests!

### Question 4

Correct

Marked out of 1.00

Flag question

#### Question text

Write a Python program which accepts a sequence of comma-separated numbers from user and generate a list and a tuple with those numbers.

For example:

Input:

13,2,4,5

Output

[13,2,4,5]

(13,2,4,5)

**For example:**

| **Input** | **Result** |
| --- | --- |
| 13,2,4,5 | ['13', '2', '4', '5']  ('13', '2', '4', '5') |

Answer:(penalty regime: 0 %)

numbers = input()

list\_of\_numbers = numbers.split(",")

tuple\_of\_numbers = tuple(list\_of\_numbers)

print(f"{list\_of\_numbers}")

print(f"{tuple\_of\_numbers}")

CheckQuestion 4

#### Feedback

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 13,2,4,5 | ['13', '2', '4', '5']  ('13', '2', '4', '5') | ['13', '2', '4', '5']  ('13', '2', '4', '5') |  |
|  | 12 | ['12']  ('12',) | ['12']  ('12',) |  |
|  | 1,2 | ['1', '2']  ('1', '2') | ['1', '2']  ('1', '2') |  |
|  | 1, 3,4,15,22,16,7 | ['1', ' 3', '4', '15', '22', '16', '7']  ('1', ' 3', '4', '15', '22', '16', '7') | ['1', ' 3', '4', '15', '22', '16', '7']  ('1', ' 3', '4', '15', '22', '16', '7') |  |

Passed all tests

### Question 5

Correct

Marked out of 1.00

Flag question

#### Question text

Write a Python function **product(lst)** to return the product of the list **lst** of integers

**For example:**

| **Test** | **Result** |
| --- | --- |
| print(product([3,4,7,11])) | 924 |

Answer:(penalty regime: 0 %)

def product(lst):

p = 1

for i in lst:

p \*= i

return p

CheckQuestion 5

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | print(product([3,4,7,11])) | 924 | 924 |  |
|  | print(product([3])) | 3 | 3 |  |
|  | print(product([3,4])) | 12 | 12 |  |
|  | print(product([3,4,7])) | 84 | 84 |  |

Passed all tests!

### Question 6

Correct

Marked out of 1.00

Flag question

#### Question text

Write a Python function **sum\_of\_cube** that takes a positive integer n and returns the sum of the cube of all the positive integers smaller than n.

**For example:**

| **Test** | **Result** |
| --- | --- |
| print(sum\_of\_cube(8)) | 784 |

Answer:(penalty regime: 0 %)

def sum\_of\_cube(n):

return sum(i\*\*3 for i in range (n))

CheckQuestion 6

#### Feedback

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | print(sum\_of\_cube(8)) | 784 | 784 |  |
|  | print(sum\_of\_cube(12)) | 4356 | 4356 |  |
|  | print(sum\_of\_cube(4)) | 36 | 36 |  |
|  | print(sum\_of\_cube(1)) | 0 | 0 |  |

Passed all tests!

1. [79767\_CO3005\_001733\_CLC](https://lms.hcmut.edu.vn/course/view.php?id=16551#section-2)
2. [CC03] Python Programming Code

# [CC03] Python Programming Code

|  |  |
| --- | --- |
| **Started on** | Monday, 15 January 2024, 9:11 AM |
| **State** | Finished |
| **Completed on** | Tuesday, 16 January 2024, 5:34 AM |
| **Time taken** | 20 hours 23 mins |
| **Marks** | 6.00/6.00 |
| **Grade** | **10.00** out of 10.00 (**100**%) |