

Câu hỏi ôn tập

Câu 1: Ai đã tạo ra ngôn ngữ python?

- a. James Gosling
- b. Denis Ritchie
- c. Guido Van Rossum
- d. Tom Cruise

Câu 2: Kí hiệu nào dùng để xác định các khối lệnh (khối lệnh của hàm, vòng lặp,...) trong Python?

- a. Dấu ngoặc nhọn { }
- b. Dấu ngoặc vuông []
- c. Thụt lề
- d. Dấu ngoặc đơn ()

Câu 3: Điều nào là quy tắc đúng khi đặt tên cho biến trong Python?

- a. Có thể sử dụng keyword làm tên biến.
- b. Tên biến có thể bắt đầu bằng một chữ số.
- c. Tên biến có thể có các ký hiệu như !, @, #, \$, %,...
- d. Tên biến có thể bắt đầu bằng dấu gạch dưới "_".

Câu 4: Output của lệnh sau là:

```
print(1, 2, 3, 4, sep='*')
```

- a. 1*2*3*4
- b. 1 2 3 4
- c. 1234
- d. 24

Câu 5: Lệnh nào dùng để lấy dữ liệu đầu vào từ người dùng?

- a. input()
- b. cin
- c. scanf()
- d. <>

Câu 6: Kết quả của đoạn code dưới đây là:

```
numbers = [2, 3, 4]  
print(numbers)
```

- a. 2,3,4
- b. 2 3 4
- c. [2 3 4]
- d. [2, 3, 4]

Câu 7: Output của lệnh là:

```
print(3 >= 3)
```

- a. 3 >= 3

- b. True
- c. False
- d. None

Câu 8: Câu lệnh sử dụng toán tử and trả về kết quả TRUE khi nào?

- a. Cả hai toán hạng đều là TRUE.
- b. Cả hai toán hạng đều là FALSE.
- c. Một trong hai toán hạng là TRUE.
- d. Toán hạng đầu tiên là TRUE.

Câu 9: Cấu trúc điều khiển while trong python:

- a. Thực hiện khối lệnh trước khi kiểm tra điều kiện
- b. Không được hỗ trợ trong python
- c. Kiểm tra điều kiện trước khi thực hiện khối lệnh
- d. Khối lệnh của cấu trúc điều khiển while bắt buộc phải sử dụng lệnh break

Câu 10: Lệnh continue trong python

- a. Chấm dứt hoàn toàn, thoát ra khỏi vòng lặp không thực hiện tiếp các vòng lặp kế tiếp.
- b. Được sử dụng để bỏ qua phần còn lại của code bên trong vòng lặp, áp dụng cho lần lặp hiện tại.
- c. Không thể sử dụng trong vòng lặp for
- d. Python không có lệnh continue

Câu 11: Toán tử 'in' và 'not in' thuộc kiểu gì trong python?

- a. Toán tử cơ bản
- b. Toán tử quan hệ.
- c. Toán tử membership
- d. Toán tử identify

Câu 12: Python hỗ trợ bao nhiêu loại biến?

- a. 2 (function, class)
- b. 4 (int, float, complex, string)
- c. 1 (struct)
- d. 3 (global, local, nonlocal)

Câu 13: Khối lệnh của hàm trong python được quy ước như thế nào?

- a. Bắt đầu bằng dòng thụt lề đầu tiên sau từ khóa 'def' và kết thúc bằng dòng không thụt lề
- b. Toàn bộ khối lệnh sau từ khóa 'def'
- c. Là toàn bộ các lệnh trong cặp dấu ngoặc nhọn '{}' sau def
- d. Là các lệnh sau từ khóa 'def' và trước từ khóa 'return'

Câu 14: Vòng lặp nào trả về kết quả dưới đây?

```
11111
22222
33333
44444
55555
```

- a. `for i in range(1, 6):`
 `print(i, i, i, i, i)`
- b. `for i in range(1, 6):`
 `print(str(i) * 5)`
- c. `for i in range(1, 5):`
 `print(str(i) * 5)`
- d. `for i in range(0, 5):`
 `print(str(i) * 5)`

Câu 15: Điền phần còn thiếu trong đoạn code để được out dưới đây:

```
55555
44444
33333
22222
11111
for i in range(5, 0, _____):
    print(str(i) * 5)
```

- a. 0
- b. None
- c. -1
- d. 1

Câu 16: Kết quả nào là output của đoạn code dưới đây?

```
x = True
y=False
z= False

if not x or y:
    print (1)
elif not x or not y and z:
    print (2)
elif not x or y or not y and x:
    print (3)
else:
    print (4)
```

- a. None

- b. 1
- c. 2
- d. 3

Câu 17: Kết quả nào là output của đoạn code dưới đây?

```
for i in range(10):  
    if i == 5:  
        break  
    else:  
        print(i)  
else:  
    print("Here")
```

- a. 0 1 2 3 4
- b. 0 1 2 3 4 Here
- c. 0 1 2 3 4 5 Here
- d. 1 2 3 4 5

Câu 18: Theo dõi đoạn code dưới đây và chọn đáp án đúng:

```
i = 0  
x = 0  
while i < 10:  
    if i % 2 == 0:  
        x += 1  
    i += 1
```

x = _____.

- a. 0
- b. 3
- c. 4
- d. 5

Câu 19: Phát biểu nào chính xác khi nói về Hàm trong Python?

- a. Hàm có thể được tái sử dụng trong chương trình.
- b. Sử dụng hàm không có tác động tích cực gì đến các module trong chương trình.
- c. Không thể tự tạo các hàm của riêng người viết chương trình.
- d. Khi xây dựng hàm bắt buộc phải có thông số truyền vào.

Câu 20: Từ khóa nào được sử dụng để bắt đầu hàm?

- a. Def
- b. Fun
- c. Define
- d. Function

Câu 21: Phát biểu nào sau đây là đúng về lệnh “return” trong python:

- a. Lệnh return chỉ có thể trả về một giá trị cho hàm
- b. Lệnh return chỉ hỗ trợ trả về nhiều giá trị cho hàm
- c. Trong python ko hỗ trợ lệnh return
- d. Trả về giá trị cho hàm

Câu 22: Hàm nào sau đây là hàm tích hợp sẵn trong Python

- a. seed()
- b. sqrt()
- c. print()
- d. factorial()

Câu 23: Hàm pow(x,y,z) được diễn giải là:

- a. $(x**y)**z$
- b. $(x**y) / z$
- c. $(x**y) \% z$
- d. $(x**y)*z$

Câu 24: Phát biểu nào là đúng khi nói về Docstring trong Python?

- a. Docstring là bắt buộc phải có trong một hàm
- b. Docstring không thể được truy cập bởi thuộc tính `__doc__` trên đối tượng
- c. Docstring là chuỗi đầu tiên ngay sau tiêu đề hàm
- d. Docstring phải được viết trên một dòng

Câu 25: Kết quả của biểu thức sau là gì?

`round(4.576)`

- a. 4
- b. 4.5
- c. 4.6
- d. 5

Câu 26: Kết quả của hàm sau là gì?

`all([2, 4, 0, 6])`

- a. 0
- b. False
- c. Error
- d. True

Câu 27: Kết quả của biểu thức sau là gì?

`round(4.5676, 2)`

- a. 4.5
- b. 4.56
- c. 4.57
- d. 4.6

Câu 28: Output của hàm sau là gì?

```
any([2>8, 4>2, 1>2])
```

- a. True
- b. Error
- c. False
- d. 4>2

Câu 29: Các kết quả của hàm hiển thị dưới đây là gì?

```
sum(2, 4, 6)
```

```
sum([1, 2, 3])
```

- a. Error, 6
- b. 12, Error
- c. 12, 6
- d. Error, Error

Câu 30: Output của hàm sau là:

```
all(3, 0, 4.2)
```

- a. True
- b. False
- c. Error
- d. 0

Câu 31: Kết quả của biểu thức sau là gì?

```
min(max(False, -3, -4), 2, 7)
```

- a. 2
- b. False
- c. -3
- d. 4

Câu 32: Output của chương trình dưới đây là gì?

```
def sayHello():  
    print('Hello World!')
```

```
sayHello()
```

```
sayHello()
```

- a. Hello World!
Hello World!
- b. 'Hello World!'
'Hello World!'
- c. Hello
Hello
- d. Không có đáp án đúng

Câu 33: Khẳng định nào là đúng khi nói về đoạn code sau:

```
def printHello():
    print("Hello")
a = printHello()
```

- a. printHello() là một hàm và a là một biến. Cả hai đều không phải đối tượng.
- b. Cả printHello() và a đều thể hiện chung một đối tượng.
- c. printHello() và a là hai đối tượng khác nhau.
- d. Lỗi cú pháp. Không thể gán hàm cho một biến trong Python.

Câu 34: Output của chương trình dưới đây là gì?

```
def say(message, times = 1):
    print(message * times)
say('Hello')
say('World', 5)
```

- a. Hello
World 5
- b. Hello
World,World,World,World,World
- c. Hello
WorldWorldWorldWorldWorld
- d. Hello
HelloHelloHelloHelloHello

Câu 35: Output của chương trình dưới đây là gì?

```
def printMax(a, b):
    if a > b:
        print(a, 'is maximum')
    elif a == b:
        print(a, 'is equal to', b)
    else:
        print(b, 'is maximum')
printMax(3, 4)
```

- a. 3
- b. 4
- c. 4 is equal to
- d. 4 is maximum

Câu 36: Output của chương trình dưới đây là gì?

```
x = 50
def func(x):
    print('Giá trị của x là', x)
    x = 2
    print('Giá trị của x được thay đổi thành', x)
```

```
func(x)
print('Giá trị hiện tại của x là', x)
```

- Giá trị hiện tại của x là 2
- Giá trị hiện tại của x là 25
- Giá trị hiện tại của x là 50
- Giá trị hiện tại của x là 100

Câu 37: Output của chương trình dưới đây là gì?

```
x = 50
def func():
    global x
    print('Giá trị của x là', x)
    x = 2
    print('Giá trị của x được thay đổi thành', x)
func()
print('Giá trị hiện tại của x là', x)
```

- Giá trị của x là 50
Giá trị của x được thay đổi thành 2
Giá trị hiện tại của x là 50
- Giá trị của x là 50
Giá trị của x được thay đổi thành 2
Giá trị hiện tại của x là 2
- Giá trị của x là 50
Giá trị của x được thay đổi thành 50
Giá trị hiện tại của x là 50
- Giá trị của x là 50
Giá trị của x được thay đổi thành 50
Giá trị hiện tại của x là 2

Câu 38: Đây là output của chương trình dưới đây?

```
def outerFunction():
    global a
    a = 20
    def innerFunction():
        global a
        a = 30
        print('a =', a)
a = 10
outerFunction()
print('a =', a)
```

- a = 10 a = 30

- b. a = 10
- c. a = 20
- d. a = 30

Câu 39: Output của chương trình dưới đây là gì?

```
def func(a, b=5, c=10):
    print('a bằng', a, 'và b bằng', b, 'và c bằng', c)
func(3, 7)
func(25, c = 24)
func(c = 50, a = 100)
```

- a. a bằng 7 và b bằng 3 và c bằng 10
a bằng 25 và b bằng 5 và c bằng 24
a bằng 5 và b bằng 100 và c bằng 50
- b. a bằng 3 và b bằng 7 và c bằng 10
a bằng 5 và b bằng 25 và c bằng 24
a bằng 50 và b bằng 100 và c bằng 5
- c. a bằng 10 và b bằng 7 và c bằng 3
a bằng 15 và b bằng 5 và c bằng 20
a bằng 100 và b bằng 5 và c bằng 50
- d. a bằng 3 và b bằng 7 và c bằng 10
a bằng 25 và b bằng 5 và c bằng 24
a bằng 100 và b bằng 5 và c bằng 50

Câu 40: Output của chương trình dưới đây là gì?

```
def maximum(x, y):
    if x > y:
        return x
    elif x == y:
        return 'Các số bằng nhau'
    else:
        return y
print(maximum(2, 3))
```

- a. 2
- b. 3
- c. 6
- d. Các số bằng nhau

Câu 41: Kết quả nào là output của đoạn code dưới đây?

```
def f(value):
    while True:
        value = (yield value)
```

```

a=f(10)
print(next(a))
print(next(a))
print(a.send(20))

```

- a. 10
None
20
- b. 10
10
- c. 10
10
20
- d. 10
None
None

Câu 42: Kết quả của chương trình được in ra màn hình là?

```

x = 1
y = "2"
z = 3

sum = 0
for i in (x,y,z):
    if isinstance(i, int):
        sum += i
print (sum)

```

- a. 2
- b. 3
- c. 4
- d. 6

Câu 43: Kết quả của biểu thức sau là gì?

```

chr('97')
chr(97)

```

- a. Error
a
- b. a
Error
- c. 'a'
a

d. Error

Error

Câu 44: Kết quả của hàm sau là gì?

`complex(1+2j)`

a. Error

b. 1

c. 2j

d. 1+2j

Câu 45: Đây là kết quả của hàm `complex()` trong những đáp án dưới đây?

a. 0

b. 0j

c. 0+0j

d. Error

Câu 46: Output của hàm biểu diễn dưới đây là gì?

`divmod(10.5, 5)`

`divmod(2.4, 1.2)`

a. (2.0, 0.5)

(2.0, 0.0)

b. (2.00, 0.50)

(2.00, 0.00)

c. (2, 0.5)

(2, 0)

d. (2, 0.5)

(2)

Câu 47: Kết quả của hàm hiển thị dưới đây là gì?

`x=3`

`eval('x^2')`

a. Error

b. 1

c. 6

d. 9

Câu 48: Output của hàm sau là gì?

`float('1e-003')`

`float('2e+003')`

a. 3.00

300

b. 0.001

2000.0

- c. 0.001
200
- d. Error
2003

Câu 49: Output của hàm sau là gì?

```
float('      -12345\n')
```

Lưu ý: số lượng khoảng trắng trước số đó là 5.

- a. -12345.0
- b. -12345.0 (5 khoảng trắng trước số)
- c. Error
- d. -12345.000000000.... (số thập phân vô hạn)

Câu 50: Kết quả của hàm hiển thị dưới đây là gì?

```
ord(65)
```

```
ord('A')
```

- a. A
65
- b. A
Error
- c. Error
Error
- d. Error
65

Câu 51: Kết quả của hàm sau là gì?

```
float('-infinity')
```

```
float('inf')
```

- a. -infinity
inf
- b. -inf
inf
- c. Error
Error
- d. Error
Junk value

Câu 52: Hàm nào sau đây sẽ không xảy ra lỗi khi không truyền tham số cho nó?

- a. min()
- b. divmod()
- c. float()
- d. all()

Câu 53: Kết quả của biểu thức sau là gì?

`hex(15)`

- a. `0xf`
- b. `f`
- c. `0xF`
- d. `0Xf`

Câu 54: Hàm nào sau đây không gây ra lỗi?

- a. `ord()`
- b. `ord(' ')`
- c. `ord("")`
- d. `ord("")`

Câu 55: Kết quả của hàm hiển thị dưới đây là gì?

`len(["hello", 2, 4, 6])`

- a. Error
- b. 3
- c. 4
- d. 6

Câu 56: Kết quả của hàm sau là gì?

`oct(7)`

`oct('7')`

- a. `0o7`
Error
- b. Error
`07`
- c. `07`
Error
- d. `07`
`0o7`

Câu 57: Cho `a = [1,2,3,4,5]` thì điều nào sau đây là đúng?

- a. `print(a[:]) => [1,2,3,4]`
- b. `print(a[0:]) => [2,3,4,5]`
- c. `print(a[-1:]) => [1,2]`
- d. `print(a[:100]) => [1,2,3,4,5]`

Câu 58: `n` trong đoạn sau là kiểu dữ liệu nào?

`n = '10'`

- a. integer
- b. tuple
- c. string
- d. operator

Câu 59: Đây là kết quả của đoạn code dưới đây?

```

number = 5.0
try:
    r = 10/number
    print(r)
except:
    print("Oops! Error occurred.")

```

- a. 2.0
- b. 5.0
- c. Oops! Error occurred.
- d. 2.0 Oops! Error occurred.

Câu 60: Đây là kết quả của đoạn code dưới đây?

```

myList = [1, 5, 5, 5, 5, 1]
max = myList[0]
indexOfMax = 0
for i in range(1, len(myList)):
    if myList[i] > max:
        max = myList[i]
        indexOfMax = i
print(indexOfMax)

```

- a. 0
- b. 1
- c. 4
- d. 5

Câu 61: Theo dõi đoạn code dưới đây và chọn đáp án đúng:

```

a = [1, 4, 20, 2, 5]
x = a[0]

```

```

for i in a:
    if i > x:
        x = i
print(x)

```

- a. x là giá trị lớn nhất của list.
- b. x là giá trị trung bình của list.
- c. x là giá trị nhỏ nhất của list.
- d. x là tổng giá trị các số trong list.

Câu 62: Kết quả của chương trình được in ra màn hình là?

```

string = "my name is x"
for i in string:
    print (i, end=", ")

```

- a. m, y, , n, a, m, e, , i, s, , x
- b. my, name, is, x,
- c. Error

Câu 63: Kết quả của hàm hiển thị dưới đây là gì?

```
list(enumerate([2, 3]))
```

- a. [(0, 2), (1, 3)]
- b. Error
- c. [(1, 2), (2, 3)]
- d. m, y, , n, a, m, e, , i, s, , x,
- e. [(2, 3)]

Câu 64: Kết quả nào là output của đoạn code dưới đây?

```
s='0123456789'
print(s[3:6])
```

- a. 345
- b. 3456
- c. 012345
- d. 0123456

Câu 65: Đối tượng dưới đây thuộc kiểu dữ liệu nào?

```
L = [1, 23, 'hello', 1]
```

- a. Dictionary
- b. Tuple
- c. Array
- d. List

Câu 66: Output của lệnh dưới đây là:

```
print "Hello World"[::-1]
```

- a. d
- b. Error
- c. Hello Worl
- d. dlroW olleH

Câu 67: Kết quả nào là output của đoạn code dưới đây?

```
list = [ 'Tech', 404, 3.03, 'Beamers', 33.3 ]
print(list[1:3])
```

- a. [404, 3.03]
- b. ['Tech', 404, 3.03, 'Beamers', 33.3]
- c. ['Tech', 'Beamers']
- d. None of the above

Câu 68: Đâu không phải là kiểu dữ liệu tiêu chuẩn trong Python?

- a. List
- b. Dictionary

- c. Class
- d. Tuple

Câu 69: Đây là giá trị của colors[2]?

```
colors = ['red', 'orange', 'yellow', 'green', 'blue',  
         'indigo', 'violet']
```

- a. Orange
- b. yellow
- c. indigo
- d. blue

Câu 70: Đây là điểm khác biệt giữa tuple và list?

- a. Dữ liệu thuộc kiểu list có thể thay đổi được, dữ liệu thuộc kiểu tuple không thể thay đổi được.
- b. Tuple nằm trong dấu ngoặc nhọn, list nằm trong dấu ngoặc vuông.
- c. Tuple nằm trong dấu ngoặc vuông, list nằm trong dấu ngoặc nhọn.
- d. Tuple là danh sách với dữ liệu kiểu la mã, list là danh sách với kiểu dữ liệu thường.

Câu 71: Đây là output của đoạn code dưới đây?

```
mylist=[1, 5, 9, int('0')]  
print(sum(mylist))
```

- a. 15
- b. 16
- c. 63
- d. Không có đáp án đúng.

Câu 72: Kết quả của chương trình được in ra là:

```
mylist=['a', 'aa', 'aaa', 'b', 'bb', 'bbb']  
print(mylist[:-1])
```

- a. [a, aa, aaa, b, bb]
- b. ['a', 'aa', 'aaa', 'b', 'bb']
- c. ['a', 'aa', 'aaa', 'b', 'bb', 'bbb']
- d. Error

Câu 73: Lệnh print in ra kết quả nào cho chương trình dưới đây?

```
list1 = [1, 3]  
list2 = list1  
list1[0] = 4  
print(list2)
```

- a. [1, 3]
- b. [1, 4]
- c. [4, 3]
- d. [1, 3, 4]

Câu 74: Chạy code sau trong Python, kết quả là:

```
num = '5'*'5'
```

- a. TypeError: can't multiply sequence by non-int of type 'str'
- b. 333
- c. 27
- d. 9

Câu 75: Đây là kết quả của đoạn code dưới đây?

```
class Foo:
    def printLine(self, line='Python'):
        print(line)

o1 = Foo()
o1.printLine('Java')
```

- a. Python
- b. line
- c. Python
- d. Java

Câu 76: Đây là output của chương trình dưới đây?

```
class Point:
    def __init__(self, x = 0, y = 0):
        self.x = x+1
        self.y = y+1

p1 = Point()
print(p1.x, p1.y)
```

- a. 0 0
- b. 1 1
- c. x y
- d. None None

Câu 77: Một lớp được thừa hưởng thuộc tính từ hai lớp khác nhau được gọi là gì?

- a. Đa kế thừa (Multiple Inheritance)
- b. Kế thừa đa cấp (Multilevel Inheritance)
- c. Kế thừa phân cấp (Hierarchical Inheritance)
- d. Kế thừa (Inheritance)

Câu 78: Kết quả của chương trình dưới đây là:

```
class Point:
    def __init__(self, x = 0, y = 0):
        self.x = x
        self.y = y
    def __sub__(self, other):
        x = self.x + other.x
```

```

        y = self.y + other.y
        return Point(x,y)
p1 = Point(3, 4)
p2 = Point(1, 2)
result = p1-p2
print(result.x, result.y)
a. 0 0
b. 1 1
c. 2 2
d. 4 6

```

Câu 79: Khẳng định nào là đúng về chương trình dưới đây?

```

class A:
    def __init__(self):
        self.a = 1
        self.__b = 1
    def getY(self):
        return self.__b

```

```

obj = A()
obj.a = 45
print(obj.a)

```

- a. Chương trình chạy bình thường và kết quả được in ra là 45.
- b. Chương trình có lỗi xảy ra vì ' __b ' là thuộc tính private, không thể truy cập được từ bên ngoài lớp.
- c. Chương trình chạy bình thường và kết quả được in ra là 1.
- d. Chương trình có lỗi xảy ra vì ' a ' là thuộc tính private, không thể truy cập được từ bên ngoài lớp.

Câu 80: Khẳng định nào là đúng về chương trình dưới đây?

```

class A:
    def __init__(self):
        self.x = 1
        self.__y = 1
    def getY(self):
        return self.__y

```

```

a = A()
a.x = 45
print(a.x)

```

- a. Chương trình có lỗi xảy ra vì ' x ' là thuộc tính private, không thể truy cập được từ bên ngoài lớp.
- b. Chương trình chạy bình thường và kết quả được in ra là 1.
- c. Chương trình chạy bình thường và kết quả được in ra là 45.
- d. Chương trình có lỗi xảy ra vì ' __y ' là thuộc tính private, không thể truy cập được từ bên ngoài lớp.