Cairo University	Cairo University Faculty of Graduate Studies for Statistical Researches **Department:** Computer Sciences **Academic Year:** 2020-2021 **Semester:** Second **Date:** 27/ 6/2021 **Level:** Diploma	

Course Title:	**Course code:** CS522	**Time:** 1.5 Hours	**Exam marks:** 100	**# Exam. Sheets:** 11 Pages
Exam. Instructions : ANSWER THE FOLLOWING QUESTIONS in Python				**Model: A**

Question: (30 Mark)

Choose the correct answer for each of the following:

1. Which of the following **assign** an **integer** value to a **variable** correctly?

   (a) x=25 ***(b)*** x='hello' ***(c)*** 2x=5 ***(d)*** 2x='hello'

---

2. Which of the following is a **loop without body**?

   (a)
   ```
   x=3
   while(x>3)
   ```

   (b)
   ```
   x=3
   while(x>3);
   ```

   (c)
   ```
   x=3
   while(x>3)
   X=x+1
   ```

   (d)
   ```
   x=3
   while(x>3):
       pass
   ```

---

3. Which of the following is a correct **comment** in python?

   (a) #comment ***(b)*** '''comment

   (c) ?comment ***(d)*** /*comment*/

---

4. What is the index number of the **last element** of a **tuple** with **30 elements**?

   (a) 30 ***(b)*** 29 ***(c)*** -1 ***(d)*** b and c

---

5. The **break** statement can be used to **exit** from

(a) for loop ***(b)*** while loop ***(c)*** if statement ***(d)*** a and b

---

6. The last printed expression is assigned to the ..... variable

(a) last ***(b)*** _ ***(c)*** var ***(d)*** exp

---

7. What is the **output** of the following:

```
t1=(2, 3, 4)
t2='a', 'b', 'c'
print(t1+t2)
```

(a) 2 3 4 a b c ***(b)*** t1+t2 ***(c)*** 2 3 4 ***(d)*** a b c

---

8. ..... can store different types of values

(a) variable ***(b)*** list ***(c)*** function ***(d)*** a and c

---

9. ..... is a collection of **unordered**, **non indexed** and **non duplicated** data

(a) variable ***(b)*** set ***(c)*** list ***(d)*** tuple

---

10. What is the **output** of the following:

```
A=25
print(a)
```

(a) error ***(b)*** 25 ***(c)*** a ***(d)*** A

Question 2: (15 Marks)

Choose the equivalent code for each of the following:

11.

```
x=3
if x>3:
   print('x>3')
else:
   print('x !> 3')
```

(a)
```
x=3
if x>3 print('x>3') else print('x!>3')
```

(b)
```
x=3
if x>3 print('x>3')    print('x!>3') else
```

(c)
```
x=3
print('x>3') if x>3      else print('x!>3')
```

(d)
```
x=3
print('x>3') if x>3      print('x!>3')    else
```

12.

```
d={'k1': 25, 'k2': 46}
for k, v in d.items():
      print(k, v)
```

(a)
```
d={'k1': 25, 'k2': 46}
for k in d:
      print(k, d[k])
```

(b)
```
d={'k1': 25, 'k2': 46}
for k in d.items():
      print(d)
```

(c)
```
d={'k1': 25, 'k2': 46}
for k in d:
      print(k)
```

(d)
```
d={'k1': 25, 'k2': 46}
for k in d:
      print(d[k)
```

13.

```
num=35
if num%2==0:
    print('even')
else:
    print('odd')
```

(a)

```
num=35
print([num%2==0]('odd', 'even'))
```

(b)

```
num=35
print([num%2==0]('even', 'odd'))
```

(c)

```
num=35
print(('even', 'odd')[num%2==0])
```

(d)

```
num=35
print(('odd', 'even')[num%2==0])
```

14.

```
alpha=('a', 'b', 'c', 'd')
for i in range(len(alpha)):
    print(alpha[i])
```

(a)

```
alpha= ('a', 'b', 'c', 'd')
for i in range(len(alpha)):
    print(i)
```

(b)

```
alpha= ('a', 'b', 'c', 'd')
for i in range(len(alpha)):
    print(alpha)
```

(c)

```
alpha= ('a', 'b', 'c', 'd')
for i in alpha:
    print(i)
```

(d)

```
alpha= ('a', 'b', 'c', 'd')
for i in range(len(alpha)):
    pass
```

15.

```
z=[]
for i in range(3):
    z[i:]=[i]
```

(a) z =**[** i **for** i in range(3) **]**

(b) z =**[for** i in range(3) i]

(c) z =**[for** i in range(3) z[i:] = [i]]

(d) z =**[for** i in range(3) [i]]

Question 3: (15 Marks)

Choose the error line number in each of the following:

16.

```
1. class A:
2.         def square(x):
3.                 return x**2
4. a=A()
5. a.square(3)
```

(a) line 1 ***(b)*** line 3 ***(c)*** line 4 ***(d)*** line 5

17.

```
1. class A:
2.         def __init__(self, x):
3.                 self.__x=x

4. class B:
5.         def __init__(self, r):
6.                 self.__r=r

7. class C(A, B):
8.         pass

9. c=C(2, 4)
```

(a) line 3 ***(b)*** line 6 ***(c)*** line 7 ***(d)*** line 9

18.

```
1. class A:
2.         def __foo(self, x):
3.                 return x*2
4. a=A()
5. a.__foo(5)
```

(a) line 1 ***(b)*** line 3 ***(c)*** line 5 ***(d)*** line 4

19.

```
1. class Person:
2.         def __init__(self, name, age):
3.                 self.name=name
4.                 self.age=age

5. p=Person()
```

(a) line 2 ***(b)*** line 5 ***(c)*** line 4 ***(d)*** line 3

20.

```
1. class Person:
2.         def __init__(self, a, b):
3.                 self.__a=a
4.                 self.__b=b
5.         def foo(self, x):
6.                 return x**2

7. p=Person(3, 5)
8. print(Person.foo(10))
```

(a) line 2 ***(b)*** line 7 ***(c)*** line 6 ***(d)*** line 5

Question 4: (40 Marks)

Choose the the correct output each of the following:

21.

```
x=2
while(x<4):
    x=x+1
    if x==4:
        continue
    print(x)
```

(a) 2 3 4 ***(b)*** 2 3 ***(c)*** 4 ***(d)*** 3

22.

```
x=1
while(x<2):
    x+=1
    print(x)
else:
    print('hello')
```

(a) 2 hello ***(b)*** 1 2 ***(c)*** 2 ***(d)*** hello

23.

```
x=set()
x.add('orange')
print(x)
```

(a) orange ***(b)*** x ***(c)*** o r a n g e ***(d)*** {'orange'}

24.

```
def myfun (a, b=4):
    print(a/b)
x, y=4, 8
myfun(b=x, a=y)
```

(a) 2 ***(b)*** no output ***(c)*** 1 ***(d)*** 0.5

25.

```
def foo(**grades):
    print(len(grades))
    print(type(grades))
foo(a=96, b=82, c=70)
```

(a) 3 Dictionary ***(b)*** 6 ***(c)*** tuple ***(d)*** 6 tuple

---

26.

```
x=lambda a: a**2
print(x(3))
```

(a) 6 ***(b)*** 3 ***(c)*** 9 ***(d)*** 5

27.

```
x=[1, 2, 3]
w=x[:]
if id(w)==id(x):
    print('same object')
else:
    print('different object')
```

(a) same object ***(b)*** different object ***(c)*** true ***(d)*** false

---

28.

```
def foo(x):
    for i in range (2):
        x[i:]=[i]
x=[]
foo(x)
print(x)
```

(a) [ ] ***(b)*** [0, 1] ***(c)*** x ***(d)*** no output

29.

```
def fun (*arg):
    print(arg[2])
fun(2, 3, 4, 5)
```

(a) 2 5 ***(b)*** 4 ***(c)*** 6 ***(d)*** [5 3 9]

30.

```
def foo(a, b):
    c=a+b
    return c
x, y=3, 2
print(foo(x, y))
```

(a) 3 ***(b)*** foo ***(c)*** no output ***(d)*** 5

---

set() creates a set.

add() adds an item to a set.

items() returns a list of dictionary's (key, value) tuple pairs.

len(x) returns the number of items in the collection x.

range(n) generates a sequence of numbers from zero to n-1.