Midterm

Due Dec 16 at 11:59pm **Points** 47 **Questions** 47
Available until Dec 16 at 11:59pm **Time Limit** 240 Minutes
Allowed Attempts 2

Instructions

Welcome to the midterm portal for CPE/EE 551!

Few important notes:

- You are allowed a maximum of **two attempts**. The highest score from these attempts will be recorded. You will be able to see correct responses after you finish the test.

- You can choose to take the test anytime between now until **midnight, 12/16**.

- You must give the **entire test in one go** (you cannot take part of the test, log off, and restart).

- This is a **timed test**. The time limit for taking the test is **two (2) hours**. The test should realistically take just about an hour or less. Please make sure you have access to stable Internet and that your laptop/device is plugged into a charge point.

- This is a **solo test.** Collaboration is not allowed.

- Test is **open book**, **open Internet** and **free access to Python shell** to test out sample code snippets.

- All questions are **multiple choice** or **short answer types**.

- As always, if anything, feel free to reach out via email: **miyengar@stevens.edu (mailto:miyengar@stevens.edu)**

Take the Quiz Again

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	9 minutes	44 out of 47

ⓘ Answers will be shown after your last attempt

Score for this attempt: **44** out of 47

Submitted Dec 12 at 2:01pm

This attempt took 9 minutes.

Question 1

1 / 1 pts

Is Python a "compiled" or "interpreted" language?

○ Compiled

● Interpreted

○ Compiled and Interpreted

○ None of the above

Question 2

1 / 1 pts

Python variables have a "type" associated with them

● True

While the types are interpreted dynamically, Python variables do have a strong type associated with them. This type is immutable at any line of code.

○ False

Incorrect

Question 3

0 / 1 pts

When creating a new variable, you must specify it's type (e.g., int x, float y etc.,)

● True

You do not have to specify type, this is interpreted

○ False

Question 4

1 / 1 pts

All variables in Python are 'labels' and 'objects'

● True

Indeed!

○ False

Question 5

1 / 1 pts

A variable type can change at any new line of code. For example, a variable "x" assigned to a value 10 (integer) can be later reassigned to a brand new type (e.g., x = "this is a string" or x = 5.3456)

● True

○ False

Question 6

1 / 1 pts

Assume a Python code that consists of 10 statements. Assume further that the first 9 statements are correct while the last statement is syntactically incorrect. When this program is run, Python will still *execute* the first 9 statements before throwing an error at the last statement.

● True

○ False

Question 7

1 / 1 pts

Hows does Python help a programmer create and maintain "scopes"

□ Curly braces "{"

□ None of the above

☑ Tabs

□ Regular braces "("

Incorrect

Question 8

0 / 1 pts

Python forces programmers to write code that is easy-to-read, unlike most other programming languages

● False

○ True

Question 9

1 / 1 pts

How does Python get rid of the semi-colon (";") that most programming languages use to mark the end of a statement?

○ Must use semi-colons

● New line creates a new statement

Question 10

1 / 1 pts

A Python list can be composed of objects of different types, e.g., L = ["lol", 2.145, 914] is actually allowed

○ This is not true

● Yes indeed

Question 11

1 / 1 pts

In a list (say "L") composed of five objects, how do you access the **fourth** element of the list

○ L[4]

○ L[5-4]

● L[3]

○ L[5]

Question 12

1 / 1 pts

Consider a list L = ['a', 'b', 'c'].

What then is the value of L*2?

○ ['aa', 'bb', 'cc']

○ ['2a', '2b', '2c']

○ This is not allowed, i.e., you cannot multiply lists like that

● ['a', 'b', 'c', 'a', 'b', 'c']

Question 13

1 / 1 pts

Any Python list can be configured to act as a Stack, Queue or a Linked-List.

○ No answer text provided.

○ No answer text provided.

○ That's not possible

● Yes indeed

Question 14

1 / 1 pts

Every Python Dictionary element must have a **key:value** pair

○ No answer text provided.

● Yes, it is required

○ No answer text provided.

○ No, it is optional

Question 15

1 / 1 pts

In Python, control structure is achieved by which of the following directives:

○ if then else

○ None of the above

● if, elif, else

○ if else-if catch

Question 16

1 / 1 pts

In a Python program, control structure:

○ Defines program-specific data structures

● Directs the order of execution of the statements in the program

○ Manages the input and output of text characters

○

Dictates what happens before the program starts and after it terminates

Question 17

1 / 1 pts

Assume **x** and **y** are labels created/set appropriately. Which of the following snippets will NOT execute correctly:

Snippet 1.

```
if(x > 0 and y < 100):
  print 'foo'
```

Snippet 2.

```
if(x > 0 and y < 100): print 'foo'
```

Snippet 3.

```
if(x > 0 and y < 100):
print 'foo'
```

Snippet 4.

```
if(x > 0 and y < 100):
         print 'foo'
```

○ Snippet 4

○ Snippet 2

○ Snippet 1

● Snippet 3

Question 18

1 / 1 pts

What signifies the end of a control block in Python?

○ }

○ A comment starting with #

○ end

● A line that is indented less than the previous line

Incorrect

Question 19

0 / 1 pts

What gets printed on screen?

```
if 'b' in 'box':
   print 1
   print 2
      if 'a' in 'apple':
          print 3
```

○ box, apple

● Prints: "1, 2, 3"

○ "1,2". There is an error with the indentation of the nested if ('a' in 'apple')

○ This code is faulty, prints nothing

Question 20

1 / 1 pts

```
What will the following snippet print?

----------------

d = {'a': 0, 'b': 1, 'c': 0}

if d['a'] > 0:
   print 'A ok'
elif d['b'] > 0:
   print 'B ok'
elif d['c'] > 0:
   print 'C ok'
elif d['d'] > 0:
   print 'D ok'
else:
   print('not ok')

------------------
```

○ the program simply fails to execute

● B ok

○ A ok

○ uncaught exception

○ C ok

○ not ok

Question 21

1 / 1 pts

Assuming 'x' and 'y' are appropriately defined variables, which of the following snippets is a VALID if/else statement in Python?

Snippet 1.

```
if x < y: if x > 10: print 'foo'
```

Snippet 2.

```
if x < y: print 'less'
elif y < x: print 'more'
else: print 'baz'
```

Snippet 3.

```
if x < y: print 'big' else: print 'small'
```

○ Snippet 3

○ Snippet 1

● Snippet 2

Question 22

1 / 1 pts

```
----------------------
def mult(a, b):
   print a * b

label_x = mult
label_x(2,3)
-----------------------

What will the snippet shown above print on terminal?
```

○ Prints nothing on screen

○ Syntax Error: This is not allowed

● 6

○ a + b

Question 23

1 / 1 pts

```
-------------------------
def outerFun(a, b):
    def innerFun(c, d):
        return c * d
    return innerFun(a, b)

res = outerFun(5, 10)
print(res)

--------------------------
What does the snippet above print on screen?
```

● 50

○ 500

○ 15

○ Syntax Error in outerFun

Question 24

1 / 1 pts

```
-------------------
def func(num):
    return num + 25

func(5)
print(num)
-------------------

What does the above snippet print?
```

○ num

○ 30

○ Prints nothing on screen

● NameError: num is undefined and out of scope

Question 25

1 / 1 pts

```
age = 22
if age > 18:
   def display_content():
      print "Showing contents in category 1"
else:
   def display_content():
      print "Showing contents in category 2"

display_content()
-----------------------

What does the snippet above print?
```

○ Showing contents in category 2

● Showing contents in category 1

○ Prints nothing on screen

○ Error: redefinition of display_content

Question 26

1 / 1 pts

```
-------------------------
def fun1(name, age=20):
    print(name, age)

fun1('Emma', 25)
-------------------------

What does the above print?
```

○ Error: You can't set age=20 in function definition

○ Emma 20

○ name age

● Emma 25

Question 27

1 / 1 pts

Python functions can return multiple values and are not limited to just one return value

● True

○ False

Question 28

1 / 1 pts

```
--------------------
def add(a, b):
  return a+5, b+5

result = add(3, 2)
print(result)
---------------------

What does the above snippet print?
```

○ 15

● (8, 7)

○ result

○ Error assigning add to result

Question 29

1 / 1 pts

Python functions always return a value, even if there is no explicit return statement in the function definition.

● True

Yes, this is true. When there is no explicit return value, Python functions return 'NaN'

○ False

Question 30

1 / 1 pts

Lambda functions make the program more 'readable', i.e., it becomes easy to collaborate and makes the programs more intuitive to read.

○ True

● False

Question 31

1 / 1 pts

What is the output of the following code snippet?

```
func = lambda x : return x
print func(2)

-----------
```

● Syntax Error

Correct! Lambda cannot 'return' a value

○ 2.0

○ x

○ 2

Question 32

1 / 1 pts

What is the output of the following code snippet?

```
(lambda x: (x + 3)*5/2)(3)
```

○ 0

● 15.0

○ Syntax Error

○ 30

Question 33

1 / 1 pts

Given a list (e.g., ['Real', 'Python']), we have a regular function that appends an argument to the list. Convert the following function (in bold-face) to a lambda:

```
my_list = ['Real', 'Python']
def func(x):
   return ''.join(x)
```

○ lambda x: return ' '.join(x)

○ Cannot convert this to a lambda

● lambda x: ' '.join(x)

○ lambda func(x): return ' '.join(x)

Question 34

1 / 1 pts

What gets printed on screen?

```
L = [lambda x : x ** 2,
     lambda x : x ** 3,
     lambda x : x ** 4]

for item in L:
   print (item(2))
```

○ Nothing at all

○ 2, 3, 4

○ Error: you cannot use item in for-loop

● 4, 8, 16

Question 35

1 / 1 pts

The "filter()" function in python accepts a function and a list as an argument, and returns a new list based on the items which evaluate as 'True' for the function.

For the snippet below, what get's printed on screen?

```
my_list = [1, 5, 4, 6, 8, 11, 3, 12]
new_list = list(filter(lambda x: (x%2 == 0) , my_list))

print new_list
```

○ [NaN]

○ Error: prints nothing

● [4, 6, 8, 12]

○ [1, 5, 4, 6, 8, 11, 3, 12]

Question 36

1 / 1 pts

Which of the following represents a template, blueprint or a contract to generate objects of the same type?

● A class

○ A method

○ A data-field

○ An object

Question 37

1 / 1 pts

Which of the following keywords marks the beginning of a class definition?

○ def

○ All of the above

○ return

● class

Question 38

1 / 1 pts

Which of the following is required to create a new instance of a class?

○ value returning method

● Constructor

○ A None method

○ print statements

Question 39

1 / 1 pts

What will be the output of the following snippet?

========

```
class Sales:
  def __init__(self, id):
    self.id = id
    id = 100

var = Sales(123)
print var.id
```

● 123

○ Error: Fails to run

○ NaN

○ 100

Question 40

1 / 1 pts

The "self" argument is implicit for every function call made by an object of a class definition

● True

○ False

Question 41

1 / 1 pts

It is possible for an object of a class to independently declare new variables that belong only to that particular object, violating the original class definition

● True

○ False

Question 42

1 / 1 pts

What the the correct "extension" for a file to contain a Python package/module?

● .py

○ .pym

○ .pymodule

○ .module

Question 43

1 / 1 pts

To bring in a module from an external file, which of the following keywords must one use?

○ load

○ None of the above

● import

○ include

Question 44

1 / 1 pts

Assume a python module called "Adder" which has a function within it called "Add". To import just this "Add" function, which of the following directives must one use?

○ import Add

● from Adder import Add

○ import Add from Adder

Question 45

1 / 1 pts

All python files are, by definition, a package ready to be used as an import

● True

○ False

Question 46

1 / 1 pts

The purpose of the "__main__" boilerplate is to ensure that python imports the definitions and not the 'action' portion

● True

○ False

Question 47

1 / 1 pts

The "import" statement shall literally inject code from another file directly onto the file that invokes the import

● True

○ False

Quiz Score: **44** out of 47