CS 1358 Introduction to Programming in Python

Spring Semester 2024

Prof. Pai H. Chou

Self-Check 2

Answer the following questions to check your understanding of your material. Expect the same kind of questions to show up on your tests.

## 1. Definitions and Short Answers

1. What is a **comment** in a program and what is its purpose?
2. What is an **operator**? Give some examples of **arithmetic operators** in Python.

+-\*/%

1. What is a **comparison operator**? What are possible results of a comparison?

>、<、>=、=<

1. What is a **logical operator**? What are possible results of a logical operation?

and or ==

1. What is 20 in **hexadecimal** representation? in **octal** representation?

0x14, 0o24

1. Why does Python support two **division operators**? What is their difference?

/ -> for not integer

// -> for integer

1. What is the difference between **'12'** and **12** in Python?

‘12’ is string

12 is integer

1. What is the difference between **x = 3** and **x == 3** in Python?

a = 3 means 3 is assign to a

a == 3 means it is checking whether a = 3 or not

1. Assuming the variable y has been assigned the integer value of 4, which of the following are legal in Python and what do they do? which are illegal in Python?
   * y = 4 yes
   * 4 = y no
   * y == 4 yes
   * 4 == y yes
   * 'y' = y no
   * 'y' == '4' yes
   * '4' = y no
2. Assume variable x has integer value 3, and variable y has integer value of 4. What is the result of the following operator expressions, if they are legal in Python? Which of the following are not legal?
   * x \* y yes
   * 'x' \* y yes
   * x \* 'y' yes
   * 'x' \* 'y' no
   * x + y yes
   * 'x' + 'y' yes
   * 'x' + y no
   * x + 'y' no
3. What is the data type of ['Sun','Mon','Tue','Wed','Thu','Fri','Sat']? List
4. if L = ['Sun','Mon','Tue','Wed','Thu','Fri','Sat'], then what are the **values** of the following **expressions** if they are legal Python? Which are illegal?
   * L[3] yes
   * L[1:5] yes
   * L[5:1] yes
   * L(2:3) no
   * L[1,2,3] no
   * L{3}no
   * L[1-5] yes
   * L['3']no
5. Assume T = ('Sun','Mon','Tue','Wed','Thu','Fri','Sat'), which of the following are allowed in Python, and what are their output or effect? Which are not allowed, for what reasons?
   * print(T[3]) yes
   * print(T(3)) no
   * print(T{3})no
   * T[3] = 'WED' no
   * T[3] == 'WED' yes
   * print(T[3:5]) yes
   * print(T[3, 5]) no
   * print(T['3']) no
6. Assume S = {'Sun','Mon','Tue','Wed','Thu','Fri','Sat'}, which of the following are allowed in Python, and what are its output or effect? Which are not allowed, for what reasons? All no
   * print(S[3])
   * print(S(3))
   * print(S{3})
   * S[3] = 'WED'
   * S[3] == 'WED'
   * print(S[3:5])
   * print(S[3, 5])
   * print(S['3'])
7. Assume D = {'Sun':0, 'Mon':1, 'Tue':2, 'Wed':3, 'Thu':4, 'Fri':5, 'Sat':6}, which of the following are legal in Python, and what are their values?
   * D[3] N
   * D['Thu'] Y
   * D[0:3] N
   * D[2, 6] N
   * D{'Sun'} N
   * D(0) N
   * D{3} N
   * D('Sun')N

What is the value of { 2, 3, 4 } | { 3, 4, 5 } ? {2, 3, 4, 5}

1. What is the value of { 2, 3, 4 } & { 3, 4, 5 }? {3, 4}
2. Suppose you have the following sequence of Python statements:  
   x = 3  
   y = 2  
   **if** x > y:  
    print("x is bigger than y")  
   **elif** x == y:  
    print("x and y are the same")  
   **else**:  
    print("x is smaller than y")  
   What is printed? x is bigger than y
3. What is wrong with the following code, which is supposed to compute the total of a list of numbers?  
   L = [3, 2, 6, 5]  
   **for** i **in** L:  
    total = total + i  
   print(total)  
   How can it be fixed? total = 0
4. What is the difference between  
   x = 0  
   **while** x < 100:  
    x = x + 1 -> x = 100  
   and  
   x = 0  
   **if** x < 100:  
    x = x + 1 -> x = 1  
   ?
5. What is an example of a **function** in Python? How do you **call** a function? What is a **parameter**?
6. What is an example of calling a (built-in) function that **returns a value**?
   * does input() return a value?
   * does print() return a value?
   * what other built-in functions do you know that returns a value?
7. Python supports two kinds of **loops**. What are they?
8. What is a **suite**? What is the pronunciation of "suite"?
9. What does **import** math do? How do you call the cos function (cosine) defined in the math module in Python?
10. To read a file, it is common to see fh = open('filename'). What kind of data is fh called? Give an example of using fh for accessing (e.g., reading or writing) a file.
11. if s = 'hello', Python supports two styles of “calls” (or “invocation”):
    * len(s) is an example of a **function call**
    * s.upper() is another form of call. What kind of call is it?
12. How are **class** and **instance** related to each other?
13. Why is it incorrect to split the statement  
    f = a + b \* 2 + c / 2 - 4 \* d  
    onto two separate lines as the following  
    f = a + b \* 2 + c / 2  
     - 4 \* d  
    ? How can it be fixed so Python will accept it?
14. If you want to **swap** the values of two variables x and y, why can't you just do  
    x = y  
    y = x  
    ?  
    Give two different ways you can swap their values correctly in Python.
15. What is a **keyword** in Python? Give some example keywords in Python.
16. Which of the following are legal and illegal **identifiers** in Python?
    * mynameY
    * my\_nameY
    * \_mynameY
    * MyNameY
    * myname\_Y
    * my-name N
    * my11name Y
    * myname11 Y
    * 11myName N
    * my\_11Name yes
    * \_11myNameyes
    * @myname no
    * my@name no
    * myname@ no
    * in no
    * out yes
    * \_in yes
    * \_out yes
    * IN yes
    * OUT yes
    * and no
    * or no
    * but yes
    * function Y
    * integer yrs
    * number yes
    * class N
    * instance Y
    * global N
    * local Y
    * you+me no
    * I\_love\_$$ no
17. What is an example of a **snake-case** identifier? a **camel-case** identifier?

snake\_case: hello\_world

camel\_case:helloWorld