**Lab 1 assignment**

Name: Yerraballi Suresh Kumar Reddy

USN:19MCAR0080

MCA

Python Programming Lab

Date: 29/01/21

**1. Is the literal 4 a valid Python expression?**

4

**Output:**

Out[1]: 4

**2. Is the variable x a valid Python expression?**

x

**Output:**

Traceback (most recent call last):

File "<ipython-input-5-6fcf9dfbd479>", line 1, in <module>

x

NameError: name 'x' is not defined

**3. Is x + 4 a valid Python expression?**

x+4

**Output:**

Traceback (most recent call last):

File "<ipython-input-6-b30e489716d6>", line 1, in <module>

x+4

NameError: name 'x' is not defined

**4. What affect does the unary + operator have when applied to a numeric expression?**

5+4

**Output:**

Out[9]: 9

**5. Sort the following binary operators in order of high to low precedence: +, -, \*, //, /, %, =.**

**Output:**

4+6/2-4\*7

Out[8]: -21.0

**6. Given the following assignment:**

x = 2

Indicate what each of the following Python statements would print.

(a) print("x")

(b) print('x')

(c) print(x)

(d) print("x + 1")

(e) print('x' + 1)

(f) print(x + 1)

**Output:**

x

x

2

x+1

**---------------------------------------------------------------------------**

**TypeError** Traceback (most recent call last)

**<ipython-input-2-7f2efcb61309>** in <module>

4 print**(**x**)**

5 print**("x+1")**

**----> 6** print**('x'+1)**

7 print**(**x**+1)**

**TypeError**: can only concatenate str (not "int") to str

**7. Given the following assignments:**

i1 = 2

i2 = 5

i3 = -3

d1 = 2.0

d2 = 5.0

d3 = -0.5

Evaluate each of the following Python expressions.

(a) i1 + i2

(b) i1 / i2

(c) i1 // i2

(d) i2 / i1

(e) i2 // i1

(f) i1 \* i3

(g) d1 + d2

(h) d1 / d2

(i) d2 / d1

(j) d3 \* d1

(k) d1 + i2

(l) i1 / d2

(m) d2 / i1

(n) i2 / d1

(o) i1/i2\*d1

(p) d1\*i1/i2

(q) d1/d2\*i1

(r) i1\*d1/d2

(s) i2/i1\*d1

(t) d1\*i2/i1

(u) d2/d1\*i1

(v) i1\*d2/d1

**Output:**

5.0

**8. What is printed by the following statement:**

#print(5/3)

**Output:**

Blank because it’s a comment

**9. Given the following assignments:**

i1 = 2

i2 = 5

i3 = -3

d1 = 2.0

d2 = 5.0

d3 = -0.5

Evaluate each of the following Python expressions.

(a) i1 + (i2 \* i3)

(b) i1 \* (i2 + i3)

(c) i1 / (i2 + i3)

(d) i1 // (i2 + i3)

(e) i1 / i2 + i3

(f) i1 // i2 + i3

(g) 3 + 4 + 5 / 3

(h) 3 + 4 + 5 // 3

(i) (3 + 4 + 5) / 3

(j) (3 + 4 + 5) // 3

(k) d1 + (d2 \* d3)

(l) d1 + d2 \* d3

(m) d1 / d2 - d3

(n) d1 / (d2 - d3)

(o) d1 + d2 + d3 / 3

(p) (d1 + d2 + d3) / 3

(q) d1 + d2 + (d3 / 3)

(r) 3 \* (d1 + d2) \* (d1 - d3)

**Result:**

52.5

**10. What symbol signifies the beginning of a comment in Python?**

#

#this is a comment

**11. How do Python comments end?**

‘’’

‘’’this is the

comments"’

**12. Which is better, too many comments or too few comments?**

Too few comments

**13. What is the purpose of comments?**

To explain(describe) or guide about a particular statement/s incase if needed

**14. Why is human readability such an important consideration?**

So that it is easier to learn, understand, and execute faster

**15. What circumstances can cause each of the following run-time errors to arise?**

• NameError

• ValueError

• ZeroDivisionError

• IndentationError

• OverflowError

• SyntaxError

• TypeError

Hint: Try some of following activities in the interpreter or within a Python program:

• print a variable that has not been assigned

• convert the string 'two' to an integer

• add an integer to a string

• assign to a variable named end-point

• experiment adding spaces and tabs at various places in the code of an error-free Python program

• compute raise a floating-point number to a large power, as in 1.510,000

**print(name)**

**---------------------------------------------------------------------------**

**NameError** Traceback (most recent call last)

**<ipython-input-15-9ba126b17b03>** in <module>

**----> 1** print**(**name**)**

**NameError**: name 'name' is not defined

num='two'

int(num)

**---------------------------------------------------------------------------**

**ValueError** Traceback (most recent call last)

**<ipython-input-16-8266ee371c8e>** in <module>

1 num**='two'**

**----> 2** int**(**num**)**

**ValueError**: invalid literal for int() with base 10: 'two'

a=20

b='hi'

c=a+b

**---------------------------------------------------------------------------**

**TypeError** Traceback (most recent call last)

**<ipython-input-19-12ce1d8d2798>** in <module>

1 a**=20**

2 b**='hi'**

**----> 3** c**=**a**+**b

**TypeError**: unsupported operand type(s) for +: 'int' and 'str'

end-point='45'

**File "<ipython-input-21-34e8b5878782>", line 1**

**end-point='45'**

**^**

**SyntaxError:** cannot assign to operator

a=1.5\*\*10000

**---------------------------------------------------------------------------**

**OverflowError** Traceback (most recent call last)

**<ipython-input-25-8e4eadd65e32>** in <module>

**----> 1** a**=1.5\*\*10000**

**OverflowError**: (34, 'Result too large')

**A=**

**File "<ipython-input-26-2d920c342ece>", line 1**

**a=**

**^**

**SyntaxError:** invalid syntax

**16. Consider the following program which contains some errors. You may assume that the comments**

within the program accurately describe the program’s intended behavior.

# Get two numbers from the user

n1 = float(input()) # 1

n2 = float(input()) # 2

# Compute sum of the two numbers

print(n1 + n2) # 3

# Compute average of the two numbers

print(n1+n2/2) # 4

# Assign some variables

d1 = d2 = 0 # 5

# Compute a quotient

print(n1/d1) # 6

# Compute a product

n1\*n2 = d1 # 7

# Print result

print(d1) # 8

**Output:**

**File "<ipython-input-43-c656ee7343a2>", line 13**

**n1\*n2 = d1 # 7**

**^**

**SyntaxError:** cannot assign to operator

**17. Write the shortest way to express each of the following statements.**

(a) x = x + 1

(b) x = x / 2

(c) x = x – 1

(d) x = x + y

(e) x = x - (y + 7)

(f) x = 2\*x

(g) number\_of\_closed\_cases = number\_of\_closed\_cases + 2\*ncc

**Output:**

x += 1

x /= 2

x -= 1

x += y

x -= (y + 7)

x \*= 2

n += 2\*n

**18. What is printed by the following code fragment?**

x1 = 2

x2 = 2

x1 += 1

x2 -= 1

print(x1)

print(x2)

Why does the output appear as it does?

**Output:**

3

1

**19. Consider the following program that attempts to compute the circumference of a circle given the**

radius entered by the user. Given a circle’s radius, r, the circle’s circumference, C is given by the

formula:

C = 2πr

r = 0

PI = 3.14159

# Formula for the area of a circle given its radius

C = 2\*PI\*r

# Get the radius from the user

r = float(input("Please enter the circle's radius: "))

# Print the circumference

print("Circumference is", C)

(a) The program does not produce the intended result. Why?

Statement is interchanged

(b) How can it be repaired so that it works correctly?

By changing into a correct place

r = 0

PI = 3.14159

# Formula for the area of a circle given its radius

r = float(input("Please enter the circle's radius: "))

C = 2\*PI\*r

# Get the radius from the user

# Print the circumference

print("Circumference is", C)

**Output:**

Please enter the circle's radius: 4

Circumference is 25.13272