

Review

by Deborah R. Fowler

updated from python 2 to 3
ie. print to print()



KEY CONCEPTS

- ✓ • variables
- ✓ • truth statements
- ✓ • looping
- ✓ • functions
- ✓ • I/O
- ✓ • lists
- classes/objects
- OOP



Today:

- E3 – recursion

recursionExampleWhat.py - C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\recursio...

File Edit Format Run Options Window Help

```
1 # Remember to include title/description/author/date in your top block comment
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10
11
12 # A simple example of recursion - similar to the factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(x,currentLevel,maxLevel):
16     print("At level = ", currentLevel, " x = ", x)
17     if (currentLevel == maxLevel):
18         return x;
19     else:
20         return testrecursion(x+1,currentLevel+1,maxLevel)
21
22
23 result = testrecursion(1,0,2)
24 print ("Result of recursion", result)
25
```



IDLE Shell 3.10.2

File Edit Shell Debug Options Window Help

```
Python 3.10.2 (tags/v3.10.2:a58ebc...
t (AMD64)] on win32
Type "help", "copyright", "credits
>>>
= RESTART: C:\Users\Deborah\Desktop\
\Class11-Quiz\recursionExamples\re
At level = 0 x = 1
At level = 1 x = 2
At level = 2 x = 3
Result of recursion 3
>>> |
```

The example from last day

```
recursionExampleWhatStep1.py - C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\re...
File Edit Format Run Options Window Help
1 # Remember to include title/description/author/date in your top block commen
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10 import random
11
12 # A simple example of recursion - similar to the factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(currentLevel,maxLevel):
16     myFunctions = ["cat()", "dog()"]
17     kermit = random.choice(myFunctions)
18     if (currentLevel == maxLevel):
19         return kermit
20     else:
21         return kermit + testrecursion(currentLevel+1,maxLevel)
22
23
24 result = testrecursion(0,2)
25 print ("Result of recursion", result)
26
```



```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022) [AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
= RESTART: C:\Users\Deborah\Desktop\SRCWebSite\Class11-Quiz\recursionExamples\recursionExampleWhatStep1.py
Result of recursion dog()dog()cat()
>>> |
```

Now we have a string being built with cat() and dog()

```
recursionExampleWhatStep2.py - C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\re...
File Edit Format Run Options Window Help
1 # Remember to include title/description/author/date in your top block comment
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10 import random
11
12 # A simple example of recursion - similar to the factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(currentLevel,maxLevel):
16     myFunctions = ["cat()", "dog()"]
17     myOperators = ["+"]
18     kermit = random.choice(myFunctions)
19     myOp = random.choice(myOperators)
20     if (currentLevel == maxLevel):
21         return kermit
22     else:
23         return kermit + myOp + testrecursion(currentLevel+1,maxLevel)
24
25
26 result = testrecursion(0,2)
27 print ("Result of recursion", result)
28
```



```
IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan
t (AMD64)] on win32
Type "help", "copyright", "credits" or "l
>>>
= RESTART: C:\Users\Deborah\Desktop\SRCWe
\Class11-Quiz\recursionExamples\recursior
Result of recursion dog()+cat()+cat()
>>> |
```

I have added an operator “+”

```

File Edit Format Run Options Window Help
1 # Remember to include title/description/author/date in your top block comment
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10 import random
11
12 # A simple example of recursion - similar to the factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(currentLevel,maxLevel):
16     myFunctions = ["cat(", "dog("]
17     myVariables = ["x", "y", "x + y", "x * y"]
18     myOperators = ["+"]
19     kermit = random.choice(myFunctions)
20     myOp = random.choice(myOperators)
21     myVar = random.choice(myVariables)
22     if (currentLevel == maxLevel):
23         return kermit + myVar + ")"
24     else:
25         return kermit + myVar + ")" + myOp + testrecursion(currentLevel+1,maxLevel)
26
27
28 result = testrecursion(0,8)
29 print ("Result ", result)
30

```



now some variables

```

IDLE Shell 3.10.2
File Edit Shell Debug Options Window Help
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on wi
n32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\re
cursionExamples\recursionExampleWhatStep3.py
Result  cat(x * y)+dog(x)+dog(x)+cat(x * y)+cat(x)+cat(x + y)+cat(x + y)+cat(y)+dog(y)
>>>

```

But we are still ending each function.

What if we nested them?

```
1 # Remember to include title/description/author/date in your top block comment
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10 import random
11
12 # A simple example of recursion - similar to the factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(currentLevel,maxLevel,endpar):
16     myFunctions = ["cat(", "dog("]
17     myVariables = ["x", "y", "x + y", "x * y"]
18     myOperators = ["+"]
19     kermit = random.choice(myFunctions)
20     myOp = random.choice(myOperators)
21     myVar = random.choice(myVariables)
22     endpar += ")"
23     if (currentLevel == maxLevel):
24         return kermit + myVar + endpar
25     else:
26         return kermit + myVar + myOp + testrecursion(currentLevel+1,maxLevel,endpar)
27
28
29 result = testrecursion(0,8,"")
30 print ("Result of recursion", result)
31
```



rather than end, let's keep track

```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\recursionExamples\re
sionExampleWhatStep4.py
Result of recursion cat(x+cat(x * y+cat(y+cat(x + y+dog(x+dog(x * y+cat(x+cat(x+dog(x + y))))))))))
>>> |
```



```
1 # Remember to include title/description/author/date in your top block comment
2 #
3 # Recursive example - what is happening
4 #
5 # Author: Deborah R. Fowler
6 # Date: Oct 13 2018
7 #
8 # Description: an example of recursion
9
10 import random
11
12 # A simple example of recursion - similar to factorial
13 # example found at https://www.python-course.eu/recursive_functions.php
14
15 def testrecursion(currentLevel,maxLevel,endpar):
16     myFunctions = ["cat(", "dog("]
17     myVariables = ["x", "y", "x + y", "x * y"]
18     myOperators = ["+"]
19     kermit = random.choice(myFunctions)
20     myOp = random.choice(myOperators)
21     myVar = random.choice(myVariables)
22     endpar += ")"
23     if (currentLevel == maxLevel):
24         return kermit + myVar + endpar
25     else:
26         return kermit + myVar + myOp + testrecursion(currentLevel+1,maxLevel,endpar)
27
28
29 result = testrecursion(0,8,"")
30 print ("Result of recursion", result)
31
```

Here is what changed:



```
Python 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\Deborah\Desktop\SRCWebSite\PythonResources\programmingPDF\Class11-Quiz\recursionExamples\re
sionExampleWhatStep4.py
Result of recursion cat(x+cat(x * y+cat(y+cat(x + y+dog(x+dog(x * y+cat(x+cat(x+dog(x + y))))))))))
>>> |
```

File Edit Format Run Options Window Help

```
# Remember to include title/description/author/date in your top block comment
#
# Recursive example - what is happening
#
# Author: Deborah R. Fowler
# Date: Oct 13 2018
#
# Description: an example of recursion
```

```
import random
```

```
# A simple example of recursion - similar to factorial
# example found at https://www.python-course.eu/python3_recursive_functions.
```

```
def testrecursion(currentLevel,maxLevel,endpar):
    myFunctions = ["cat(", "dog("]
    myVariables = ["x", "y", "x + y", "x * y"]
    myOperators = ["+"]
    kermit = random.choice(myFunctions)
    myOp = random.choice(myOperators)
    myVar = random.choice(myVariables)
    endpar += ")"
    if (currentLevel == maxLevel):
        return kermit + myVar + endpar
    else:
        return kermit + myVar + myOp + testrecursion(currentLevel+1,maxLevel,endpar)
```

```
result = testrecursion(0,8,"")
print "Result of recursion", result
```

rather than end each one,
let's keep track until the
end

Python 2.7.14 Shell

File Edit Shell Debug Options Window Help

Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:25:58) [MSC v.1500 64 bit (AMD64)] on win32

Type "copyright", "credits" or "license()" for more information.

>>>

```
RESTART: D:\SCAD_ALL\BuildWebSite\SRCWebSite\PythonResources\programmingPDF\Class11-15-Houdini\Class11\recur
onExampleWhatStep4.py
```

```
Result of recursion cat(x+dog(x + y+cat(y+dog(x + y+cat(x+cat(x + y+cat(x * y+dog(x * y+cat(x))))))))))
```

```
>>> |
```

The string we are returning but the
endpar is tracked in the parameter

Next steps – replace cat/dog with cos/sin

- Use values π * in your sin/cos calls
- Consider adding functions such as sqrt
- You can get interesting patterns even at 4 levels of recursion and 8 works well

In-class

- Continue to work on Exercise 3
 - Use values `*pi` in your `sin/cos` calls

KEY CONCEPTS

- ✓ • variables
- ✓ • truth statements
- ✓ • looping
- ✓ • functions
- ✓ • I/O
- ✓ • lists
- classes/objects
- OOP

KEY CONCEPTS

Continue to keep up with your reading in the online textbook

If any of these key concepts are not clear – see me!