

Code Sleuth

Instructions

This exercise challenges you to figure out how operators and functions work.

- First, you'll see tables with code examples and their outputs. Use *these to infer what the operators and functions do.*
- Then, use your *inferences to predict what the outputs will be* for the subsequent code examples.

Code Sleuth activities show up repeatedly in EDS 221, so this exercise will also familiarize you with the format.

Hey I thought this was a programming class - why are we using pen and paper?

Great question! Reading code and reasoning about it without a computer helps learning to code because:

1. Pen and paper exercises have fewer distractions compared to an IDE, freeing your cognitive attention to focus on code itself (Sana, Weston, & Cepeda, 2013)
2. Reading code and tracing how it executes will help you create a more accurate mental model of how code executes, so you'll encounter fewer frustrations and write better code later (Xie *et al.*, 2019)
3. Predicting what code does *before* running it improves recall and deepens conceptual connections (Brod 2021)

References

- Brod, G. (2021). Predicting as a learning strategy. *Psychonomic Bulletin & Review*, 28(6), 1839-1847.
- Sana, F., Weston, T., & Cepeda, N. J. (2013). Laptop multitasking hinders classroom learning for both users and nearby peers. *Computers & Education*, 62, 24-31.
- Xie, B., Loksa, D., Nelson, G. L., Davidson, M. J., Dong, D., Kwik, H., ... & Ko, A. J. (2019). A theory of instruction for introductory programming skills. *Computer Science Education*, 29(2-3), 205-253.

New operators

Inference

Code example	Output
<code>(0:5) ^ 2</code>	<code>[1] 0 1 4 9 16 25</code>
<code>(0:5) %% 2</code>	<code>[1] 0 1 0 1 0 1</code>
<code>(0:5) %/% 2</code>	<code>[1] 0 0 1 1 2 2</code>

What do each of the operators do?

`^` **Exponent**

`%%` **Remainder (e.g. $7 \% 2$ is 1 because $7 = 2 * 3 + \underline{1}$)**

`%/%` **Remainder (e.g. $7 \% \% 2$ is 3 because $7 = 2 * \underline{3} + 1$)**

Prediction

Using the operator precedence rules you inferred above, what will be the output of the following expressions?

Code example	Output
<code>16 ^ (1/2)</code>	
<code>16 %% (1:5)</code>	
<code>16 %/% (1:5)</code>	

Order of operations

Inference

Code example	Output
1 : 3 ^ 2	[1] 1 2 3 4 5 6 7 8 9
1 : 3 * 2	[1] 2 4 6
8 / 5 %% 3	[1] 4
4 + 3 * 2	[1] 10

Order these operators from highest to lowest precedence.

- :

2
- ^

1
- * or /

4
- + or -

5
- %% or %/%

3

Prediction

Using the operator precedence rules you inferred above, what will be the output of the following expressions?

Code example	Output
1:3 * 2 ^ 2	
10 ^ (0:2)	
10 ^ 0:2	

Summary functions

Inference

Code example	Output
<code>mean(c(2, 3, 7, 1, 2))</code>	<code>[1] 3</code>
<code>length(c(2, 3, 7, 1, 2))</code>	<code>[1] 5</code>
<code>sum(c(2, 3, 7, 1, 2))</code>	<code>[1] 15</code>

Based on the examples above, what do each of the summary functions do?

`mean()` **Average value**

`length()` **Number of values**

`sum()` **Additive total of values**

Prediction

Using the rules you inferred above, what will be the output of the following expressions?
Pay close attention to the positions of the parentheses!

Code example	Output
<code>mean(1:3)^2</code>	
<code>length(1:3^2)</code>	
<code>sum((1:3)^2)</code>	

String functions

Inference

For the code examples below, assume the variable `corals` has the values "Porites", "Pocillopora", "Acropora".

Code example	Output
<code>nchar(corals)</code>	[1] 7 11 8
<code>paste(corals, collapse = ", ")</code>	[1] "Porites, Pocillopora, Acropora"
<code>substr(corals, start = 2, stop = 4)</code>	[1] "ori" "oci" "cro"

Based on the examples above, what do `nchar()`, `paste()`, and `substr()` do?

`nchar()` - number of characters in each text string

`paste()` - combines text strings together

`substr()` - a shorter text string (substring) within the text string

Prediction

Using the rules you inferred above, what will be the output of the following expression? Assume the variable `nearshore_ecosystems` has the values "coral reef", "kelp forest", "mussel bed".

Code example	Output
<pre>paste(substr(nearshore_ecosystems, start = 1, stop = 2), collapse = "-")</pre>	

How would the output of `nchar(nearshore_ecosystems)` differ from the output of `length(nearshore_ecosystems)`?

`nchar()` would be 3 values, one for each text string in `nearshore_ecosystems`

`length()` would be 1 value, describing the number of text strings in `nearshore_ecosystems`