

UNIVERSITY OF CALIFORNIA, BERKELEY
Department of Electrical Engineering and Computer Sciences
Computer Science Division

CS10 Summer 2026

TA: Nahee Jeong



Discussion-4: HOFs and Lambdas

Instructions

- If you're attending this section in-person, please log your attendance!
- If you missed this discussion, fill out this entire worksheet (optional portions will be titled), and upload it to the Gradescope assignment titled "Discussion-3" before the next Discussion.
- Please open up snap.berkeley.edu/run on your computer.

Group Activity / Question of the Day

- What is your favorite Berkeley restaurant? Do you have any go-to snacks?

Q1. Higher-Order Functions

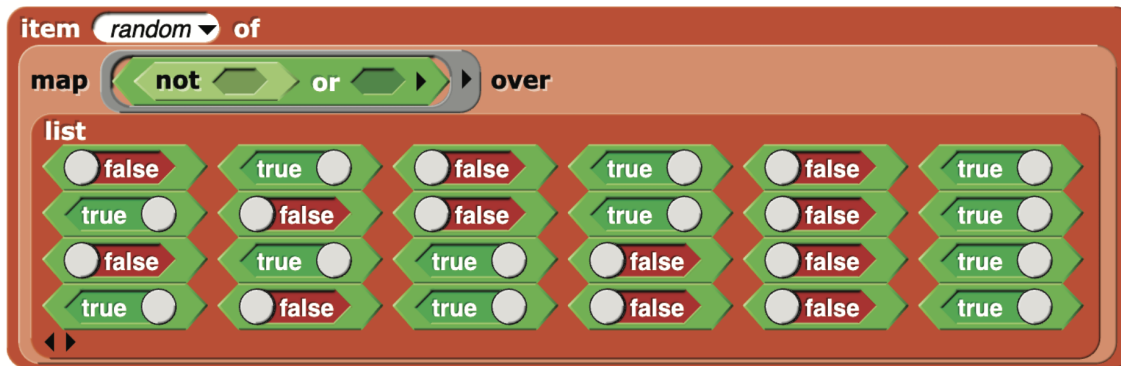
First, your TA will do a mini-review on Lists and HOFs. As you watch the presentation, you'll be able to answer the following questions.

Problems:

1. What's a *higher-order-function*? What are the main HOFs that we'll use in CS10?
2. What do the *map*, *keep*, and *combine* blocks do? What are their domains and ranges?
3. Which of the HOFs could return a list with the same number of items as the input list? Lesser items than the input list? More items than the input list?
4. Do these HOFs mutate the input lists?
5. Select all the HOFs:

- ☐ **add 2 or 4 to** 
- ☐ **combinations**   
- ☐ **find first item**  **in** 
- ☐ **tell**  **to**  
- ☐ **script variables**  
- ☐ **test**  **w/input(s)**   **expecting output** 
- ☐ **test**  **w/inputs**   **expecting output**  

6. The following block reports _____:



- ☐ True
- ☐ False
- ☐ It errors, because we can't pass in predicates in map blocks
- ☐ It depends - the output could be True or False, since selection is random.

7. What would the following blocks report? If Error, write "ERROR".

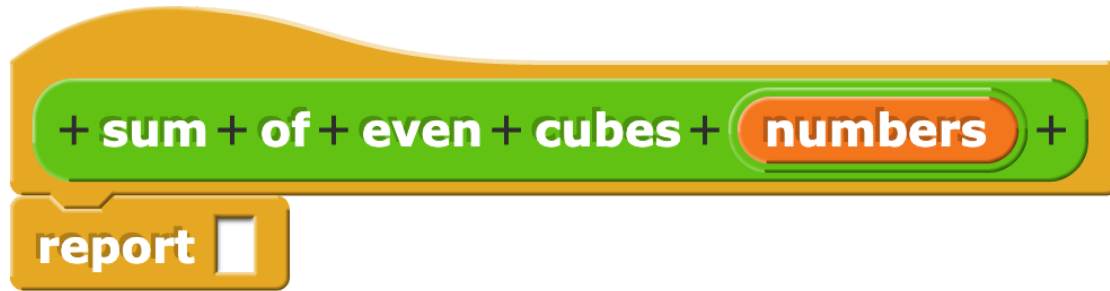


8. Complete this function. Using at least one HOF, write a reporter block, titled **sum of even cubes** that takes in a list of numbers and returns the sum of all the even numbers after they've been cubed (n^3).



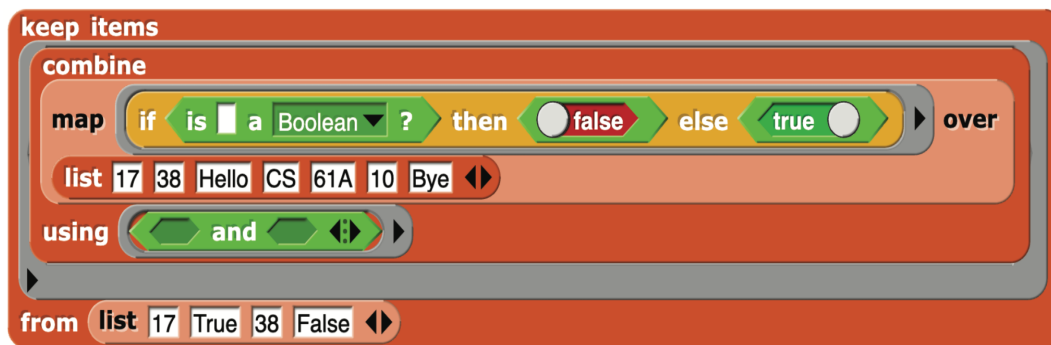
72

Sum of even cubes given the list [1, 2, 3, 4] cubes reports 72, because $2^3 + 4^3 = 8 + 64 = 72$

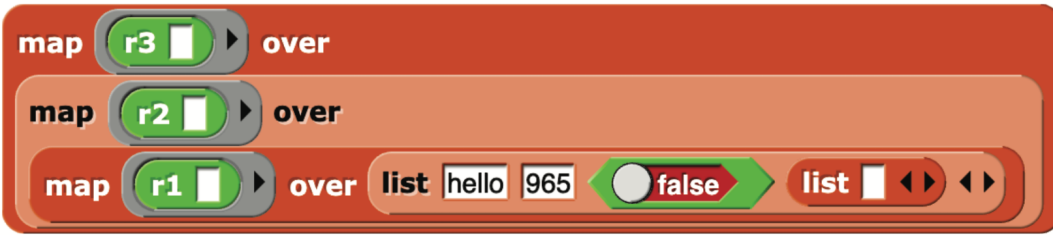


Try to do this in Snap! on your laptops.

9. What would the following block report? Use square-bracket notation for lists.



10. Answer the question based on the HOF below.



This nested block reports a list containing items.

If the block will cause an error, write 'E'.

If the answer depends on the actual reporters r1, r2, and/or r3, then write a '?'.

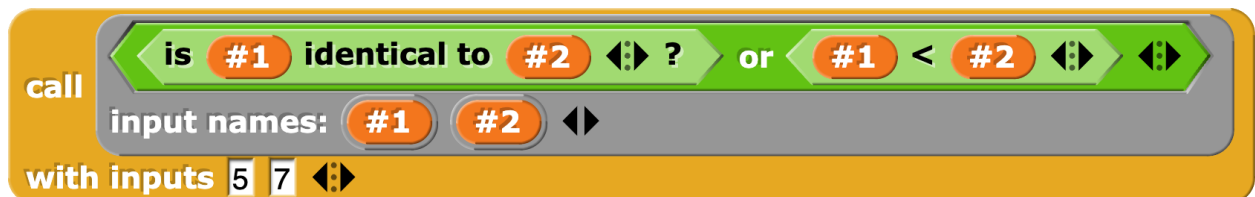
Q2. Lambdas

First, your TA will do a mini-review on Lambdas. As you watch the presentation, you'll be able to answer the following questions.

1. What does a lambda represent in Snap!?
2. How do we define a lambda in Snap!?
3. True or False: Lambdas in Snap! can be passed into higher-order functions like map?
4. Why would we use a lambda instead of a regular block/function?
5. What will this report?



6.



7. Challenge problem: Create a function, that reports the first word in a list that contains the given letter. If no words contain the letter, the function should return nothing. Do not use loops or conditional (if) blocks. Instead, use a lambda expression and higher-order

functions such as **keep items**  **from** , **item**  **1** **of** , or

find first item  **in**  to solve the problem.



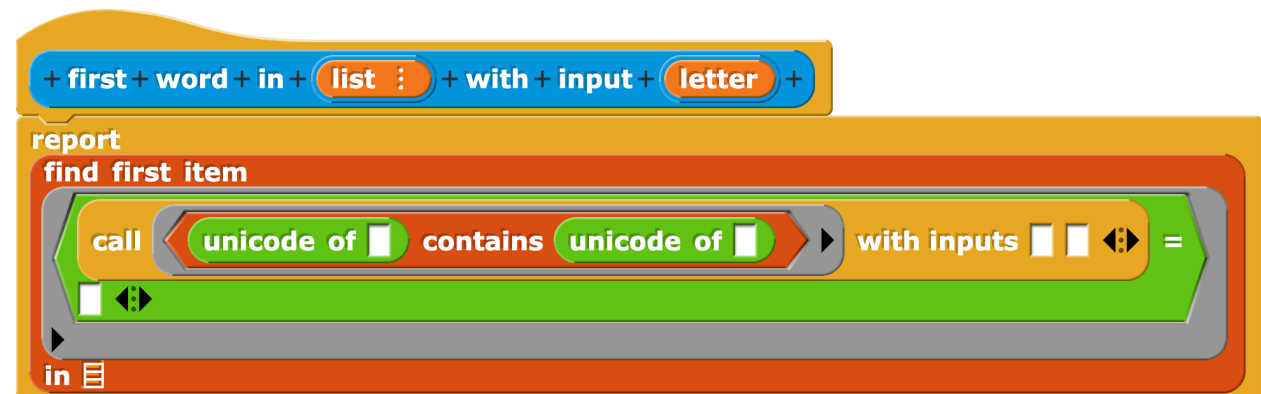
A Scratch block with a blue body and orange top and bottom. The body contains the text "+ first + word + in + list : + with + input + letter +". Below the body is an orange "report" block with a small white square input field.

What would go inside the report?



A large empty rectangular box with a black border, intended for the user to write the code that will be reported by the block above.

Alternatively, fill in the blanks so the block runs as described above.



A Scratch block with a blue body and orange top and bottom. The body contains the text "+ first + word + in + list : + with + input + letter +". Below the body is an orange "report" block. Inside the report block is a "find first item" block. The "find first item" block has a green body and contains a "call" block. The "call" block has a green body and contains the text "unicode of  contains unicode of  with inputs   = . The "find first item" block also has a green body with a small white square input field and a "in" block with a list icon.