ABMs in Philosophy:
Programming Exercises 1

Instructions: Open Netlogo. In the interface, create a button called "setup." Click on the code tab and begin the exercises below. To run the program setup, click on the Interface tab, and click your setup button.

Exercise 1: Working with strings

1. Using the WORD and ITEM functions, finish the following program by defining (i) a variable called concatenation that is equal to the string "Superman is awesome" and (ii) a variable called letterp that is equal to the string "p." Also, print the two variables in the command center.

```
to setup
   let s "Superman"
   let i "is "
   let a "awesome."
   ...
end
```

Output:
Superman is awesome.
p

2. Find three functions in the Netlogo dictionary section on strings so that the output of the program below is as the output that follows it:

```
to setup
   let mystring "Superman"
   print [1st-function-name] mystring
   print [2nd-function-name] "e" mystring
   print [2nd-function-name] "f" mystring
   print [3rd-function-name] 1 mystring "oa"
   ...
end
```

Output:
8
true

false
Soaperman

3. Predict what will happen when you run the following program; write your prediction down. Then execute the program in Netlogo. What happens? Explain what happens if your prediction differed from what occurred.

```
to setup
   print item 8 "Superman"
end
```

Exercise 2: Arithmetic Operations

In the Interface, create two different inputs. For the first, enter X as the global variable and select "Number" from the dropdown list called "Type." For the second, enter Y as the global variable and again select "Number".

1. Using the functions $+$, $\wedge$, MOD, write a setup program that prints $x + y$, x^y, and x *mod* y. For example, if you enter 2 for x and 7 for y, the output of your program should be

   Output:

   The sum of 2 and 7 is 9.
   2 to the 7th power is 128.
   7 mod 2 is 1.
   2 mod 7 is 2.
   7/2 is 3.5.

   Note: The point of this exercise is to ensure that you know how to program basic arithmetic operations. So please do not worry if your program prints out "1 to the 2th power is 1" depending upon input.

2. Pick some mathematical function from the "Mathematical" section of the Netlogo dictionary and output the result of that function when applied to x, y, or both x and y. For example, if a user enters 3.5 and 2 for x and y y respectively, and if you were to pick floor and apply it to x, your program should output:

   Output:

   The sum of 3.5 and 2 is 5.5.
   3.5 to the 2th power is 12.25

3.5 mod 2 is 1.5.
2 mod 3.5 is 2.
The floor of 3.5 is 3.

Exercise 3: Working with Boolean Variables

1. Predict what the output of the following program is. Write your prediction down and explain it. Then execute the program in NetLogo. What happens? If your prediction was not correct, explain what you think the program is doing.

```
to setup
  print 5 < 3
end
```

2. In the Interface, create two number inputs x and y just like you did in Exercise 2. Next, create a third input z and select "String" from the dropdown menu. Using functions from previous exercises and the functions and, or, and not, Write a program with the following output when a user enters 2 for x, 3 for y, and "cat" for z.

   Output:

   It is false that 2 > 3.
   It is true that the word "cat" contains at least 3 letters.
   It is false that both 2 < 3 and "cat" contains the letter "b".
   It is true that either 2 < 3 or "cat" contains the letter "b".

Exercise 4: Working With Lists In each of the following, you will employ some function on lists to complete the program below:

```
to setup
  let mylist n-values 10 [?]
  ...
end
```

1. Using the fput and lput functions, add -1 to the front of mylist and 10 to the end of mylist. Print mylist to the command center. The output of your program ought to be:

   Output:
   [-1 0 1 2 3 4 5 6 7 8 9 10]

2. Using n-values, change mylist so that each of its elements is doubled. So mylist ought to be the list below. Print mylist to the command center.

   Output:
   [-1 0 1 2 3 4 5 6 7 8 9 10]
   [-2 0 2 4 6 8 10 12 14 16 18 20]

3. Using the filter and mod functions, change mylist so that it contains only multiples of three. Print mylist to the command center. The output of your program ought to be as follows:

   Output:
   [-1 0 1 2 3 4 5 6 7 8 9 10]
   [-2 0 2 4 6 8 10 12 14 16 18 20]
   [0 6 12 18]

4. Above, you have been working with lists of numbers. In Netlogo, the elements of a list can be any data type. That is, you can have a list of numbers, booleans, strings ... and even lists themselves! Using the n-values function and only one line of code, create and print a list of length 5 such that element i of the list is the first i whole numbers. In other words, create and print the following list:

   Output:
   [[] [0] [0 1] [0 1 2] [0 1 2 3]]

5. Consider the list of lists you created in the previous exercise. Write a one-line program that prints the second item (i.e. the item with index one) in the fourth list (i.e. the list with index 3). That is, the fourth list is [0 1 2], and the second item in this list is 1.