Lightbot Functions

**Goal:** The purpose of this exercise is to learn how to express a function symbolically. You will write functions and a program to demonstrate your understanding of using these concepts.

This assignment is a continuation of the last assignment when you programmed the Lightbot symbolically, with text instructions instead of iconographic instructions.

We also introduced iteration (repeating operations), as in `4:(S)`

**Symbolic Function Definitions**

When we define functions symbolically we use a special form. For the function definition, we write the name, a pair of parentheses, the operation sequence, and end it with a period. For example, this is a function definition for a function that turns the bot around:

```
F.turnaround() R,R.
```

The *name* of the function is the part between the dot and the open parenthesis, in this case, `turnaround`. The part after the closing parenthesis and before the period is called the *body*. It defines what the function does.

To make the function happen, we need to "call" it. To call the function you give the name, followed by the parentheses, as in `turnaround()`, which instructs the bot to perform the instructions in the body of the function.

For example, to program the bot to "turn around and jump", we would write the program:

```
turnaround(), J.
```

Notice that a single function like `turnaround()` has two roles: in one role it is defined; in the other role it is called. A function has only one definition, but can be called many times.
Example: The Moon Walk

The bot version of the Moonwalk function is below.

    F.moonwalk() 4:(S, R).

According to the function body, the bot's Moonwalk is four repeats of taking a step (and not going anywhere) and then turning right. To use the moonwalk() function after jumping up two steps, for example, we could write:

    J, J, moonwalk().

Exercises

A: Consider a different solution to problem (C) from Symbolic Lightbot. Write out the definition of a function light_a_pair() such that the program 7:light_a_pair(). solves the problem shown below.

![Diagram of a board with lighted boxes]

B: The program below ends with the instruction 4:light_a_side(). Write the F.light_a_side function definition so that the program works. Your function will probably take about seven instructions.

    J, L, J, R, 4:light_a_side().

![Diagram of a board with a robot and lighted boxes]
Moonwalk

C: The Lightbot wants a new solution to the Basic Level 6 (shown below) where it goes up each riser and does the Moonwalk defined previously on the top of the riser before powering the light.

Define a new function below called `stairMoonwalk()`, where the bot goes up the stairs and does the moonwalk:

Now write out a program below that solves the entire level using your new `stairMoonwalk()` function:

To Turn In

Scan pages 2 and 3 of this document and upload them to Lightbot Functions assignment on Canvas.