Name: _____

Assessment
Light Show Animation

1) What does the state "01" represent for the lights in the tutorial?
    a) LOW
    b) OFF
    c) MEDIUM
    d) HIGH

2) How many bits are used to encode the state of one light?
    a) 1
    b) 3
    c) 4
    d) 2

3) If you wanted to create a message that says "HI" on the Light Show, what is the first step you should take according to the tutorial?
    a) Assemble the hardware
    b) Clear the screen
    c) Wire up the Light Show
    d) Create the code block

4) What should you do if the light message flashes too quickly to read?
    a) Increase the brightness
    b) Use more colors
    c) Create a new message
    d) Slow it down by adding wait time

5) What does "end-to-end encryption" mean in the context of the Light Show tutorial?
    a) The message is visible to everyone between the sender and the receiver.
    b) The message can be modified by anyone who intercepts it.
    c) The message is coded by the sender and can only be decoded by the intended receiver.
    d) The message is only visible when the lights are on.

6) Explain how the concept of bits and states in the Light Show relates to how computers process information. Give an example of another device that uses a similar method.

Name: _____

7) Describe a real-world scenario where you might want to send a coded message using lights. How would the process from the tutorial help you accomplish this task?

8) Look at the Python code you have created using Blockly for your Light Show. The code currently makes a pattern that displays the letter "H" for one second, followed by another pattern. Can you change the code so that it displays the letter "I" for one second instead of the second pattern? Explain what changes you made to the code to achieve this.

Name: _____

Answer Key
Light Show Animation

1) A - Low

2) D - 2

3) B - Clear the screen

4) D - Slow it down by adding wait time

5) C - The message is coded by the sender and can only be decoded by the intended receiver.

6) *Example:* The concept of bits and states in the Light Show is similar to how computers use binary code to process information. Each bit can be either 0 or 1, which is how computers store and manipulate data. For example, a digital clock uses binary code to represent time, where each segment of the display can be turned on or off to show different numbers.

7) *Example:* A real-world scenario for sending a coded message using lights could be maritime signaling, where ships use lights to communicate at night. The tutorial's process helps by teaching how to create and control light patterns, allowing the creation of specific signals that can be interpreted by other ships. By programming different light sequences, ships can effectively send messages like "SOS" or navigational instructions.

8) *Example:* To change the code to display the letter "I" instead of the second pattern, I need to modify the grid inside the `screen.draw(lightshow.Pix.from_grid(...))` function. Here's the updated code:

```
## ---- Imports ---- ##
import piper_lightshow as lightshow
from piper_blockly import *
import time

## ---- Definitions ---- ##

lightshow.init()
screen = lightshow.Pix()

try:
  set_digital_view(True)
except:
```

```
  pass


## ---- Code ---- ##
screen.box(0, 0, 0, 8, 8)
screen.draw(lightshow.Pix.from_grid((
  ( 0, 12, 12,  0,  0, 12, 12,  0),
  ( 0, 12, 12,  0,  0, 12, 12,  0),
  ( 0, 12, 12,  0,  0, 12, 12,  0),
  ( 0, 12, 12, 12, 12, 12, 12,  0),
  ( 0, 12, 12, 12, 12, 12, 12,  0),
  ( 0, 12, 12,  0,  0, 12, 12,  0),
  ( 0, 12, 12,  0,  0, 12, 12,  0),
  ( 0, 12, 12,  0,  0, 12, 12,  0),
)), (1-1), (1-1))
lightshow.show(screen)
send_lightshow_buffer(screen.buffer)
time.sleep(1)
screen.box(0, 0, 0, 8, 8)
screen.draw(lightshow.Pix.from_grid((
  ( 0,  0, 12, 12, 12, 12,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0,  0, 12, 12,  0,  0,  0),
  ( 0,  0, 12, 12, 12, 12,  0,  0),
)), (1-1), (1-1))
lightshow.show(screen)
send_lightshow_buffer(screen.buffer)
```

Explanation:
1. I replaced the second pattern grid with a new grid that forms the letter "I".
2. The new grid is defined as:

```
( 0,  0, 12, 12, 12, 12,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0,  0, 12, 12,  0,  0,  0),
( 0,  0, 12, 12, 12, 12,  0,  0),
```

3. This grid represents the letter "I" by turning on the appropriate lights in the 8x8 grid.