Name: ___________________

Assessment
Game Controller

1) What does a shift register do in the game controller?
    a) Sends signals to the LED
    b) Reads inputs all at the same time and shifts them out to the microcontroller
    c) Changes the color of the wires
    d) Connects the controller to the internet

2) What happens when the LATCH pulse occurs?
    a) The Pico turns off
    b) The LED changes color
    c) The Pico reads all of the pressed buttons as 1s and 0s
    d) The shift register resets

3) What is the purpose of the LED in the arrow mode function?
    a) To change the color of the LED
    b) To indicate that the Pico is off
    c) To turn on when the UP button is pressed and off when released
    d) To send signals to the shift register

4) How do you switch from arrow mode to mouse mode using the controller?
    a) Press button 5
    b) Press the UP button
    c) Press the LED
    d) Press button 11

5) What happens when you press button 5 in mouse mode?
    a) It simulates a left-mouse click
    b) The cursor moves up
    c) The cursor moves down
    d) It switches back to arrow mode

6) Explain how you could modify the controller to include an additional button that performs a new function. Describe what the new function could be and how you would implement it in the code.

Name: ____________________

7) How could you use the game controller you built in a real-world application? Provide an example and explain how it would work.

8) Look at the Python code you have written for your game controller. Can you think of a way to add a new feature where pressing button 12 on the controller increases the speed of the mouse movement? Write the new function or changes you would make to the code and explain how it works.

Name: ____________________

Answer Key

Game Controller

1) B - Reads inputs all at the same time and shifts them out to the microcontroller

2) C - The Pico reads all of the pressed buttons as 1s and 0s

3) C - To turn on when the UP button is pressed and off when released

4) D - Press button 11

5) A - It simulates a left-mouse click

6) *Example:* To add an additional button to the controller, I first need to define a new check controller block for the new button in the code and assign it a specific function. For example, if the new button is button 15, I could make it toggle to "Turbo Mode," which makes the mouse move faster. You would add code to check if button 15 is pressed and, if so, increase the mouse movement speed. This would involve adding new if statements and possibly a new variable to track the mode.

7) *Example:* The game controller could help individuals with physical disabilities who have difficulty using a traditional keyboard and mouse. For example, the controller could be adapted to work with accessible software that allows users to navigate a computer interface or play games more easily. By mapping specific buttons to common computer functions (like clicking, scrolling, and typing), the controller provides a customizable and user-friendly way for people with limited mobility to interact with technology.

8) *Example:*

```
# Additional variable to track if button 12 is pressed
BUTTON_12_pressed_state = False

# Modify the mouse_mode function to include the new feature
def mouse_mode():
  global mode_pressed
  global __speed
  flash_LED()
  flash_LED()
  while True:
    buttons_pressed_value = piper_controller.readButtons()
    __move_y = 0
    __move_x = 0
    try: __speed
```

Name: ________________

```
    except NameError: __speed = 4

    if __speed < 4 or __speed == None:
      __speed = 4
    if (piper_controller.wasPressed(BUTTON_3)):
      __move_x = __speed
    if (piper_controller.wasPressed(BUTTON_2)):
      __move_y = -1 * __speed
    if (piper_controller.wasPressed(BUTTON_1)):
      __move_x = -1 * __speed
    if (piper_controller.wasPressed(BUTTON_4)):
      __move_y = __speed
    if __move_x == 0 and __move_y == 0:
      __speed = 4
    else:
      __speed = __speed + 0.5
      mouse_HID.move(int(__move_x), int(__move_y), 0)
    if piper_controller.wasPressed(BUTTON_5) and BUTTON_5_pressed_state == False:
      mouse_HID.press(Mouse.LEFT_BUTTON)
      BUTTON_5_pressed_state = True
    elif not (piper_controller.wasPressed(BUTTON_5)) and BUTTON_5_pressed_state ==
True:
      mouse_HID.release(Mouse.LEFT_BUTTON)
      BUTTON_5_pressed_state = False
    if piper_controller.wasPressed(BUTTON_6) and BUTTON_6_pressed_state == False:
      mouse_HID.press(Mouse.RIGHT_BUTTON)
      BUTTON_6_pressed_state = True
    elif not (piper_controller.wasPressed(BUTTON_6)) and BUTTON_6_pressed_state ==
True:
      mouse_HID.release(Mouse.RIGHT_BUTTON)
      BUTTON_6_pressed_state = False
    if piper_controller.wasPressed(BUTTON_14) and BUTTON_14_pressed_state == False:
      keyboard_HID.press(Keycode.ESCAPE)
      BUTTON_14_pressed_state = True
    elif not (piper_controller.wasPressed(BUTTON_14)) and BUTTON_14_pressed_state
== True:
      keyboard_HID.release(Keycode.ESCAPE)
      BUTTON_14_pressed_state = False
    if piper_controller.wasPressed(BUTTON_11):
      mode_pressed = True
    if mode_pressed and not (piper_controller.wasPressed(BUTTON_11)):
      mode_pressed = False
      break

    # New code to increase speed with BUTTON_12
    if piper_controller.wasPressed(BUTTON_12) and BUTTON_12_pressed_state == False:
      __speed += 1
      BUTTON_12_pressed_state = True
    elif not (piper_controller.wasPressed(BUTTON_12)) and BUTTON_12_pressed_state
== True:
      BUTTON_12_pressed_state = False
```

Explanation: In this modified code, a new button (BUTTON_12) is added to increase the speed of the mouse movement. The state of BUTTON_12 is tracked using `BUTTON_12_pressed_state`. When BUTTON_12 is pressed, the speed (`__speed`) is increased by 1. The state is then set to True to ensure the speed only increases once per press. When the button is released, the state is set back to False, allowing future presses to increase the speed again.