## Lesson 16 – Activity Sheet

## Getting Started

The example program shows you how to measure and respond to **acceleration** along the *x*-axis. This measures the micro:bit being tilted to either the left or the right.

from microbit import \*

while True:

reading = accelerometer.get\_x()

if reading > 20:

display.show("R")

elif reading < -20:

display.show("L")

else:

display.show("-")

The *y*-axis relates to the micro:bit being tilted forwards or backwards. Adapt the program to measure these movements. Remember that the value 20 can be adjusted.

from microbit import \*

while True:

reading = accelerometer.get\_y()

if reading > 20:

display.show("R")

elif reading < -20:

display.show("L")

else:

display.show("-")

Now adapt your program to measure and respond to the *z*-axis. Remember this the up and down movement Again you may need to adjust the values depending on how fast you move the micro:bit.

## Success Criteria

* The micro:bit responds to acceleration on one of the axes, *x*, *y* or *z*
* The micro:bit responds to acceleration on a combination of **two** of the axes, *x*, *y* or *z*
* The micro:bit responds to acceleration on all three axes
* The responses are standard, LED or scrolling text message

## Pro-tip

The responsiveness of the micro:bit depends on how fast you move it. Try to stay consistent in movements so that you can test the code and set the values at a suitable level.

## Test Time

**Testing needs to be conducted safely**. Be careful when moving the micro:bit with pace, check who is behind you or in front. Keep adjusting the values until the measurements and responses are accurate.

## Stretch Tasks

* micro:bit responds to acceleration on 2 axes
* micro:bit responds to acceleration on all 3 axes

## Final Thoughts

This has been a program that has combined acceleration of three axes. You have also had to test your program and adjust the values to create responses that work in the real world.