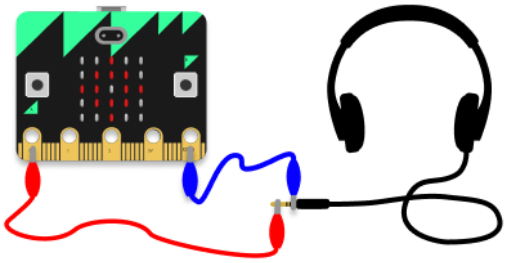
Extension: Play that funky music!

Games are better with sounds!

Use headphone and the music library to make your game more exciting!



Task 1.1: Set up the headphones!

First, we'll need to connect our headphones like in the picture above:

NOTE: you can skip this step if you have a V2 micro:bit (shiny gold logo). Your micro:bit already has a speaker built-in.

1. Connect one alligator clip to the **GND** pin of the **micro:bit**. Connect the other end to the **base** of your headphone jack.
2. Connect another alligator clip to **pin 0** of the **micro:bit**. Connect the other end to the **tip** of your headphone jack.

Task 1.2: Play a sound!

Let's play the A tone when you need to press A!

1. At the top of your code, `import music`.
2. Inside the `if` statement that checks to see if `"press a"` was selected, play the tone `"A"` for **two** beats.
3. Make sure that you set **wait** to `False` so the game keeps running while the music is playing!

Hint - Playing sounds

To play a G tone for 5 beats, you can use the following code:

```
music.play("G:5")
```

Let's make the other actions play sounds too!

1. Inside the `if` statement that checks to see if `"press b"` was selected, play the tone `"B"` for `two` beats. Make sure that `wait` is set to `False`.
2. Do the same thing for any other actions you have, making sure that they each have a unique tone!

Task 1.4: Let's listen

Test your code!

1. Can you hear all the different sounds? Make sure you test every action!

☑ CHECKPOINT ☑

If you can tick all of these off you have finished this Extension:

- ☐ When `"press a"` is the selected action, the A tone plays for 2 beats.
- ☐ When `"press b"` is the selected action, the B tone plays for 2 beats.
- ☐ For all the other actions you have, a unique sound is played for 2 beats.
- ☐ You can hear the sounds through your headphones!

★ BONUS 1.5: Make it talk ★

What if our Bop It could talk!

Speech is a lot like music, but we can tell it to say words!

Once you import the speech library you can start telling it things to say:

```
import speech
speech.say("BOP!")
```

Challenge: Can you make it announce the move each turn?