Engineering Team,

We will be using the CuteBot and Micro:Bit to model Ampere's systems. This will be done largely based on the fact that the same type of technology will be implemented in the final product for ease of acquisition and assembly. We need you to develop the software for the LED headlights. This is a critical component that needs to work 100% of the time, every time. Proper illumination allows the driver, and in some cases cameras, to be aware of what is directly in front of the vehicle at any given moment. Lights also inform others what the driver of the car is intending to do, like using a turn signal or the brake lights illuminating when braking.

At your workstation, you should have the following:

- Project Envelope
- Watt Corp standard issue laptop (with charger)
- Micro:Bit
- CuteBot car
- 3x AAA Batteries
- 1x Micro-USB to USB-A cable

Please take a moment to familiarize yourself with the equipment around you. When you are ready, open Mu-Editor.

When a car is driving, it is important that it understands certain system states. For example, when a button is pressed, lights turn on.

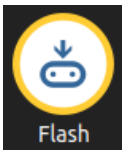
In the new file that Mu opened, write the following:

```
from microbit import *

display.show(Image.HAPPY)

while True:
    if button_a.is_pressed():
        display.show(Image.SAD)
    elif button_b.is_pressed():
        display.show(Image.SILLY)
    else:
        display.show(Image.HAPPY)
```

Click the save button! It should ask where and what you want to save this file as. You should save it in:

/home/<your username>/mu_code

Name your file: main.py

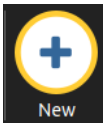


Next, hook up your Micro:Bit to your PC and click the flash button! In the bottom left corner of Mu Editor, it should tell you if the flashing process failed or succeeded.

If you need help with this step because your computer isn't working, ask us :)

Now, push the reset button on the back of your Mirco:Bit. You should see a smiling face. When you push button A, your Micro:Bit should be sad. When you push button B, your Micro:Bit should stick it's tongue out at you! These are what we call ***STATE CHANGES*** and are critical for electronic systems to operate.

But…the headlight LED's on my car don't work. Let's make them turn on!



In Mu Editor again, click the new button (**do not close your other file**). We are going to create a ***LIBRARY FILE***. Just like how a normal library contains books that can be checked out, a library file with contains code snippets that you can checkout and use in other pieces of code without having to re-type it again. We are going to create a library called cutebotcar.py, which will have all of our function definitions that make the Ampere drive.

In your new file, type the following:

(Look on next page)

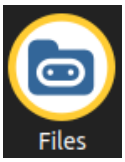
```
from microbit import *

# --- BUILD I2C DRIVERS AND CONSTANTS ---
I2CAddr = 16

# HEADLIGHTS
leftLED = 4
rightLED = 8
headlightState = False

def ledHelper(LED: int, headlightState: bool):
    buffer = bytearray(4)

    # Which LED are we setting - LEFT or RIGHT?
    buffer[0] = LED

    # If the headlights are on ...
    if headlightState:
        # ... set RED, GREEN, and BLUE to MAX
        buffer[1] = 255
        buffer[2] = 255
        buffer[3] = 255
    else:
        # ... otherwise, set RED, GREEN, and BLUE to MIN
        buffer[1] = 0
        buffer[2] = 0
        buffer[3] = 0

    # Now send the package to the address!
    i2c.write(I2CAddr, buffer)

def toggleHeadlights():
    global headlightState

    headlightState = not headlightState

    # Use the function we just made to set each LED
    ledHelper(leftLED, headlightState)
    ledHelper(rightLED, headlightState)
```

The library file now has functions that allow us to control the headlights on our car now! But…let's first save our file before we do anything else.

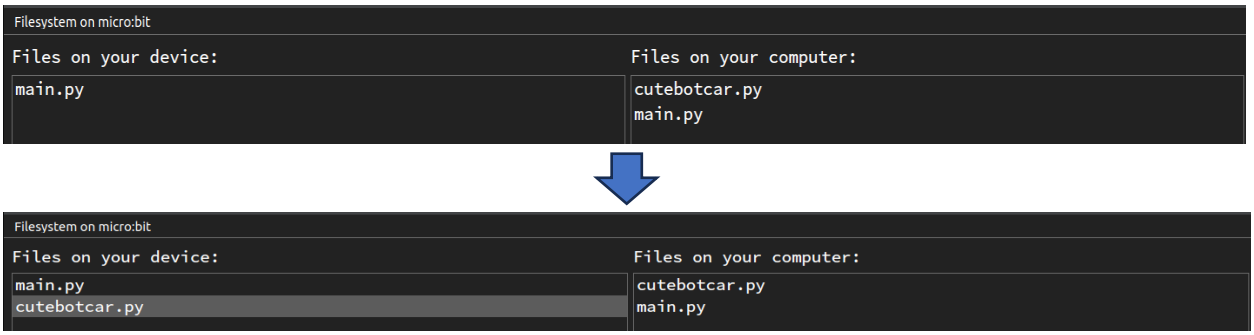
Click the save button! It should ask where and what you want to save this file as. You should save it in:

`/home/<your username>/mu_code`

Name your file: `cutebotcar.py`



We need to put our library file onto our car! Click the Files button! A window shoud pop up at the bottom of your screen. Find `cutebotcar.py` on the right and drag and drop it on the left. When the process is complete, `main.py` and `cutebotcar.py` should be on the left of the window. Click the files button again to close the window.

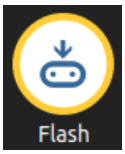


Now, open your `main.py` file again. We need to import the library we made, similar to how we imported the `microbit` library.

Everything in light grey you have already typed – so don't worry about that. Worry about everything that is in color or black.

```
from microbit import *
import cutebotcar as car

display.show(Image.HAPPY)

while True:
    if button_a.is_pressed():
        display.show(Image.SAD)
    elif button_b.is_pressed():
        display.show(Image.SILLY)
        car.toggleHeadlights()
        sleep(1000)
    else:
        display.show(Image.HAPPY)
```

Click the save button! It is important that your edits to main.py get preserved so the car can operate as intended!



Next, put your Micro:Bit in your car and hook up your Micro:Bit to your PC. Then click the flash button! In the bottom left corner of Mu Editor, it should tell you if the flashing process failed or succeeded.

If you need help with this step because your computer isn't working, ask us :)

Your Micro:Bit should still make faces at you when you push the buttons, but now it will also turn on the headlights! This is perfect for the Ampere – well done!

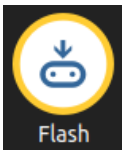
BONUS ROUND

Let's make these headlights automatically turn on if it gets too dark! We humans love automated systems, especially when it makes our life easier.

In main.py, add the following (everything in light grey you have already typed):

```
from microbit import *
import cutebotcar as car

minLight = 5
display.show(Image.HAPPY)

while True:
    light = display.read_light_level()
    if light <= minLight:
        car.toggleHeadlights()
    elif (light > minLight) and car.headlightState:
        car.toggleHeadlights()

    if button_a.is_pressed():
        display.show(Image.SAD)
    elif button_b.is_pressed():
        display.show(Image.SILLY)
        car.toggleHeadlights()
        sleep(1000)
    else:
        display.show(Image.HAPPY)
```

Click the save button! It is important that your edits to main.py get preserved so the car can operate as intended!



Next, put your Micro:Bit in your car and hook up your Micro:Bit to your PC. Then click the flash button! In the bottom left corner of Mu Editor, it should tell you if the flashing process failed or succeeded.

If you need help with this step because your computer isn't working, ask us :)

Now your car should have automatic headlights! Congratulations! The Ampere is well under-way: go talk to the managerial team to turn in your code!