ECE 476/676 - Test #1: Name ____________________

1) **Hardware: Binary Output** A 100 Watt LED requires the following

- Vf = 34V
- Id = 3000mA
- 9000 Lumens @ 3A

Design a circuit so that a Pi-Pico can turn on and off this LED with one of its binary outputs at 3 Amps. Note that the output of a Pi-Pico is

- Von = 3.3V, Iout < 12mA

If you need to make assumptions about the hardware you are using, state the assumptions you're making

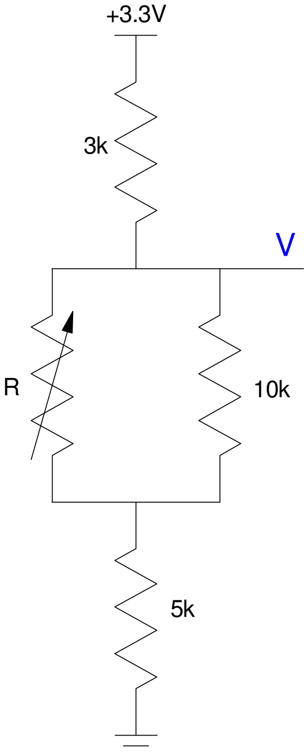
2) Hardware: Analog Inputs Design a circuit which converts x (a -5V to +10V analog signal) to y (a 0V to +3.3V analog signal)

- -5V in produces 0V out
- +10V in produces +3.3V out
- Proportional inbeteen

$$y = \left(\frac{3.3V}{15V}\right)x + \left(\frac{3.3\cdot 5}{15}\right) = 0.22x + 1.1$$

3) Python Subroutines: Write a Python subroutine which

- Is passed the temperature in degrees C, and
- Returns the voltage output for the following circuit.



Assume the thermistor has the temperature - resistance relationship of

$$R = 3000 \cdot \exp\left(\frac{4000}{T+273} - \frac{4000}{298}\right)\Omega$$

$$R_a = \left(\frac{1}{R} + \frac{1}{10k}\right)^{-1}$$

$$R_b = R_a + 5k$$

$$V = \left(\frac{R_b}{R_b+3k}\right) \cdot 3.3V$$

```
# Start of subroutine
def Voltage(T):
```

4) Python Programming Assume the hardware is set up so that a Pi-Pico can drive a 100W LED:

- GP16 = 1 (3.3V): LED is on (9000 Lumens)
- GP16 = 0 (0V): LED is off (0 Lumens)

Write a Python program adjusts the light's brightness based upon which button is pressed:

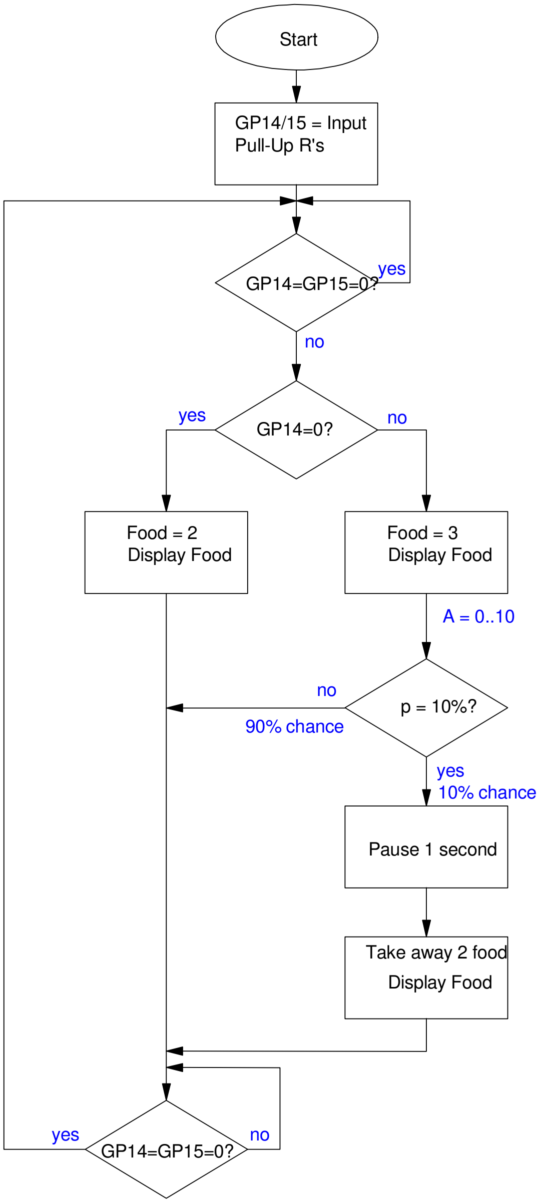
- GP0: Light Off
- GP1: Light On (1%)
- GP2: Light On (10%)
- GP3: Light On (100%)

Use whatever method you like to vary the light's brightness

5) Python Programming: A Pico is to control a mechanism which gives out food.

- When GP14 is pressed, two pieces of food are given out (Food = 2)
- When GP15 is pressed, three pieces of food are given out (Food = 3). However, 10% of the time the mechanism will then pause for one second, then take away two pieces (resulting in Food = 1)
- The program then waits for the buttons to be release and it starts over

Write the corresponding Python program



Generally Useful Python Routines

Binary Input (Button Pressed)

```
from machine import Pin

Button = Pin(15, Pin.IN, Pin.PULL_UP)
x = Button.value()
```

Binary Output (Blinking Light)

```
from machine import Pin

LED = Pin(16, Pin.OUT)
LED.toggle()
LED.value(1)
LED.value(0)
```

Analog Input (A2D Read)

```
from machine import ADC

a2d0 = ADC(0)
x = a2d0.real_u16()
```

Analog Output (PWM Output)

```
from machine import Pin, PWM

Aout = Pin(16, Pin.OUT)
Aout = PWM(Pin(16))
Aout.freq(1000)

# 0% duty cycle
Aout.duty_u16(0x0000)

# 100% duty cycle
Aout.duty_u16(0xFFFF)

# 50us pulse
Aout.duty_ns(50_000)
```

Measure a pulse width in micro-seconds

```
from machine import Pin, time_pulse_us

X = Pin(19, Pin.IN, Pin.PULL_UP)
low = time_pulse_us(19, 0, 500_000)
high = time_pulse_us(19, 1, 500_000)
```

Pause 1.23 seconds

```
from time import sleep

sleep(1.23)
```

For Loops

```
for i in range(0,6):
    d1 = i
    for j in range(0,4):
        d2 = j
        y = d1 + d2
```

While Loops

```
t = 0
while(t < 5):
    t = t + 0.01
    print(t)
```

If - else if - else statements

```
if(x < 10):
    a = 1
elif(x < 20):
    a = 2
else:
    a = 3
```

Random Numbers

```
from random import randrange

x = randrange(10)
# x = 0 to 9
```

Measure time since reset

```
from time import ticks_us

x0 = ticks_us()
```