**5. óra: Szerelem mérő**

1. 5 perces beszámoló tetszőleges tartalommal és üzenettel valamint tetszőlegesen választott retorikai eszközökkel. Nekünk kell eldönteni, hogy mi lehetett az üzenet. Figyelni kell a beszéd felépítésére (bevezető, fő rész és konklúzió) és arra, hogy a tartalom minden eleme támogassa az üzenetet, illetve, hogy használjunk retorikai eszközöket.
2. 3 perces véleményalkotás  
   - bevezető, fő rész, konklúzió  
   - 2 pozitív és egy konstruktív javaslat  
   - konklúzióban összefoglalni mi lehetett az üzenet, volt-e retorikai eszköz használva és adni még egy pozitív megjegyzést.
3. Analóg hőmérséklet mérése raspberryvel:
   1. *Bevezető*:  
      Kitaláltad, hogy megtréfálod az ismerőseidet és egy olyan eszközt raksz össze, ami képes megállapítani, hogy valaki szerelmes-e vagy sem. Természetesen van ebben egy kis turpisság, de ezt nem kell az ismerősöknek elárulni.

A történet szerint készítesz egy eszközt ami képes az emberek szerelem rezgéseit mérni, ami csak azoknak van akik szerelmesek. Az alanynak meg kell fognia az eszközt és az LED diódákon keresztül kimutatja, hogy szerelmes-e az illető. Zöld, sárga és piros kijelzők vannak. Ha a zöld világít csak, akkor az illető nem szerelmes, ha a sárga is bekapcsol, akkor az illető szerelmes de ha még a piros is ég akkor bizony fülig szerelmes.

* 1. *Előkészületek*:  
     Készítsünk ki próbapaneleket, jumper kábeleket, 10 kOhm ellenállásokat a thermisztornak, NTCC 10k thermisztorokat, piros, sárga és zöld LEDeket, hozzájuk tartozó ellenállásokat kb. 200 Ohm és az MCP3008 ADCt. Emellé szükség lesz mérőműszerre is.   
     - Rakjuk össze az áramkört. Teszteljük le műszerrel, hogy kb. 1.6 Vot mérünk-e a thermisztor lábain (feltéve, hogy szobahőmérséklet van, kb. 25 fok). Mindenesetre, amit mérünk a lábakon, azt kell majd az MCP3008al is mérnünk.   
     - Teszteljük le a műszerrel, hogy működnek-e illetve pythonból is, hogy jó e az áramkör.  
     - Teszteljük le az MCP3008at is pythonból, hogy azt olvas-e amit a műszer.   
       
     *Részletekre bontás:*Elkészíteni az áramkört.
     + Inicializálni az MCP3008at és a LEDeket.
     + Definiálni egy függvényt, ami a mért feszültség függvényében ki és be kapcsolja a LEDeket. Bemenő paraméter a feszültség és egy listába csomagolva a 3 LED.
     + Végtelen *while* ciklusban meghívni a függvényt.
  2. *Elektronika:*  
     - Készítsük el az áramkört. Érdemes kihasználni a közös földelést és tápot, hiszen sok dróttal dolgozunk majd.
  3. *Kód:*  
     - Egeszítsük ki a lenti kódvázlatot.  
     - gyakoroljuk az *if-else* szerkezetet és a többszörös feltételeket.  
     - Törekedjünk arra, hogy ha van rá mód akkor *for* ciklusban kezeljük a LEDek működését.
  4. *Tesztelés:*  
     - teszteljük a thermisztorokat a műszerrel, illetve teszteljük a programot.
  5. *Fejlesztés:*  
     - Minden egyes LED felvillanásná írjunk ki egy üzenetet a képernyőre pl. Húú, forró, forró, forróóóó....

- Írjuk át a kódot, hogy a feszültség csökkenésével aludjanak el a LED-ek. Tehát alap állapotban az összes világít.

Kód minta:

# importald a gpiozero csomagbol az MCP3008, LED objektumokat

mcp = # inicializald az MCP3008 objektumot a 7es csatornaval

red = # inicializald a piros LEDet

yellow = # inicializald a sarga LEDet

green = # inicializald a zold LEDet

def lightning(volt, leds=[green, yellow, red]):

    if .........:   # ha a volt erteke nagyobb mint 1.8

        for led in leds:

            # kapcsold ki a LEDeket

    elif volt <= 1.8 and volt > 1.5:

        leds[0].on()  # zold LED bekapcsolava EZT A SORT NEM KELL EDITALNI

        # sarga LED kikapcsol

        # piros led kikapcsol

    elif ...........:   # ha a volt <= 1.5 es volt > 1.3

        # zold LED bekapcsolva

        # sarga LED bekapcsol

        # piros led kikapcsol

    elif ...........:  # ha volt <= 1.3

        # minden LED bekapcsol

while True:

    lightning(mcp.voltage, leds=[green, yellow, red])