

Kapitel 7 : micro:bit med Python

Övning 3: Ljussensorn

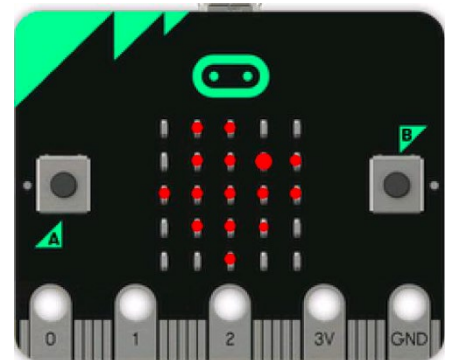
I den här lektionen kommer du att mäta ljusintensiteten med ljussensorn och spara data i en TI-Nspire-lista för vidare analys.

**Mål :**

- Att läsa av ljussensorn
- Att visa ljusstyrkan på displayen
- Att utbyta data mellan Python och TI-Nspire CX
- Att analysera data

**Lärartips:** Ljusintensiteten från en ljuskälla är omvänt proportionell mot kvadraten på avståndet från ljuskällan. Vi kommer att mäta intensiteten varje sekund. Om vi flyttar micro:bit-enheten med en konstant hastighet från ljuskällan så är avståndet proportionellt mot tiden, så tiden blir ett bra indikation på avstånd.

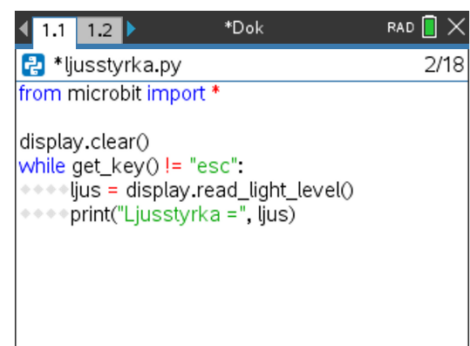
1. Micro:bit-enheten kan mäta intensiteten av omgivande ljus med hjälp av displayens lysdioder. Dessa lysdioder kan därför också användas som inmatningsenhet. (Sök gärna på "leds as light sensors")



2. Starta ett nytt Python-program i ett nytt dokument. (Tryck på **[Home]** -knappen, välj Nytt och sedan Lägg till Python> Nytt...). Ge den ett namn, till exempel "ljusstyrka". Lägg sedan till:

```
from microbit import *

display.clear()
while get_key() != "esc":
    ljus = display.read_light_level()
    print("Ljusstyrka =",ljus)
```



Metoden `display.read_light_level()` kan hittas med:

**[menu]> Fler moduler> BBC micro: bit> Sensorer**

Metoden `display.clear()` som du hittar under **[menu]> Fler moduler> BBC micro: bit>Display>Methods** släcker alla lysdioder i displayen.

- Kör programmet och rikta mikro:bit-skärmen mot en ljuskälla. Du kommer då att se ljusstyrkan på din enhet. Om du till exempel lägger handen över displayen kommer värdet att minska. Tryck på [esc] för att stoppa programmet.

*Tips: Blixen på en mobiltelefon går ofta att använda som en lampa.*

- Skapa två tomma listor före while -satsen:

```
tid = []  
ljusstyrka = []
```

Vi låter t vara en variabel för tiden:

```
t = 0
```

I den första listan ska vi lagra vid vilka tidpunkter mätningarna gjordes. I den andra listan ska vi lagra uppmätta värden.

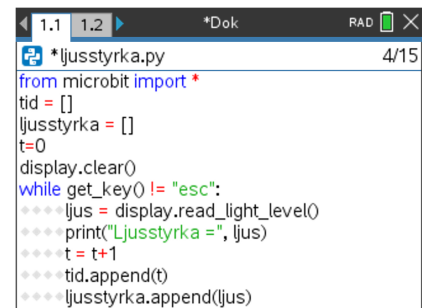
- Infoga en sats som ökar t med 1 inne i while -slingan. Lägg också till två kommandon för att lägga till t och ljusstyrka i listorna:

```
t = t+1  
tid.append(t)  
ljusstyrka.append(ljus)
```

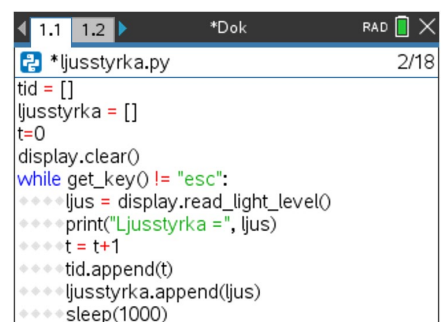
Metoden `.append` finns under **[menu]> Inbyggda> Listor**

- Tiden mellan mätningar är lite godtycklig. Vi ska fixa det genom att göra en mätning varje sekund. Du kan göra detta genom att lägga till ett `sleep`-kommando i slutet av while-loopen. `sleep(1000)` skapar en paus på en sekund.

Du hittar det här kommandot under **[menu]>BBC micro:bit>Commands**



```
*ljusstyrka.py 4/15  
from microbit import *  
tid = []  
ljusstyrka = []  
t=0  
display.clear()  
while get_key() != "esc":  
    ljus = display.read_light_level()  
    print("Ljusstyrka =", ljus)  
    t = t+1  
    tid.append(t)  
    ljusstyrka.append(ljus)
```



```
*ljusstyrka.py 2/18  
tid = []  
ljusstyrka = []  
t=0  
display.clear()  
while get_key() != "esc":  
    ljus = display.read_light_level()  
    print("Ljusstyrka =", ljus)  
    t = t+1  
    tid.append(t)  
    ljusstyrka.append(ljus)  
    sleep(1000)
```

7. När vi stoppar mätningarna med [esc] så ska vi spara Python-listorna i två TI-Nspire-listor med samma respektive namn. Du kan använda kommandot **store\_list()** för detta. Observera att dessa två kommandon inte är indragna eftersom de inte ska köras varje gång. Först när loopen är avslutad måste listorna sparas.

**store\_list()** kan hittas under:

**[menu]>BBC micro:bit>Commands**

8. Kör programmet och börja med micro:bit-enheten nära en ljuskälla. Flytta sedan långsamt enheten lite längre och längre bort från ljuskällan tills ljusstyrkan är 0.
9. När du har tillräckligt med mätningar trycker du på **[esc]** för att stoppa programmet. Lägg sedan till en extra sida i ditt dokument med **[ctrl]-[doc]** och välj **Data & statistik**. Du kommer då att se en skärm som liknar den till höger

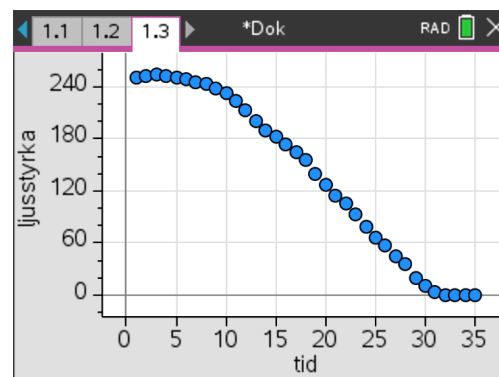
10. Klicka längst ned på skärmen för att välja en variabel och välj "tid" som en oberoende variabel. Klicka sedan på vänster sida av skärmen och välj "ljusstyrka" som den beroende variabeln. Punkterna bildar sedan en graf som förhoppningsvis någorlunda liknar den som visas här bredvid. Den exakta formen beror på hur du flyttade ljuskällan och micro:bit-enheten från varandra.

**Tips:** Om du kör experimentet igen kan du få fönstret att passa inläst data genom att välja **[menu]>Window/Zoom>Zoom Data**

```

1.1 1.2 *Dok RAD
*ljusstyrka.py 12/16
t=0
display.clear()
while get_key() != "esc":
    ljus = display.read_light_level()
    print("Ljusstyrka =", ljus)
    t = t+1
    tid.append(t)
    ljusstyrka.append(ljus)
    sleep(1000)
store_list("tid", tid)
store_list("ljusstyrka", ljusstyrka)

```



**Lärartips:** Vi såg här något som liknar förhållandet  $\text{intensitet} = k / (\text{avstånd} ** 2)$  i grafen vi fick. Du kan undersöka detta lite bättre genom att inte mäta varje sekund, utan istället på bestämda avstånd. Till exempel kan du pausa programmet med `get_key(1)`, som väntar på en tangenttryckning, och sedan flytta micro:bit-enheten en bit, till exempel 10 cm. Varje gång du sedan trycker på en knapp görs en ny mätning.