

## **Opsummering af undervisningsforløb til modul 1 efterår 2019 – LydKIT**

Lavet en interaktiv opstilling til alle grupper, som kan bruges i projektforsløb de kommende uger.

Microbit med en række input funktioner forbundet til en BeoCreate/Raspberry PI.

Der laves programmer til at reagere på knaptryk og andre sensorinputs, som medfører at forskellige lyde mixes og afspilles. Undervejs kommer vi ind på:

- Basale programmerings teknikker i Python, og et integreret udviklingsmiljø (IDE) der kan bruges på PC/MAC såvel som på Raspberry PI i opstillingerne. (Thonny). I fik det installeret på jeres egen PC/MAC via: <https://thonny.org/>
- At kunne køre programmer uden tilsluttet skærm og tastatur (headless).
- Program eksempler til at bygge videre på til jeres projekt: <https://github.com/TPSoundhub/LKM1>

Nu kan I selv arbejde videre med opstillingen og bruge den til projektopgaven. I kan gå dybere i programmering, hvis I har tid og lyst. Dels på egne PC/MAC derhjemme og dels på opstillingerne i skolen.

De forskellige inputs som kan fås fra den udleverede Micro:Bit kan I se beskrevet på det udleverede print af koden fra Microbitten i kommentarerne. Koden er også på jeres PC/MAC i LKM1 folderen. MEN I kan ikke køre den kode på PC'en. Den skal loades i MicroBitten. Det kan man også gøre fra Thonny IDE'en, men det har vi/I ikke prøvet endnu. Det er til et senere undervisningsforløb. I nuværende projekt skal I bare bruge den udleverede Micro:bit med den kode som er på den.

Mere om biblioteket Pygame kan I finde her:

<https://www.pygame.org/docs/>

Det er i dette projekt lyd og mixer delen som er brugt så fokus er på det I finder beskrevet i

<https://www.pygame.org/docs/ref/mixer.html#pygame.mixer.Sound>

I kommer langt med at bygge videre på de eksempler vi gennemgik sammen på klassen. Se kode eksemplerne i folderen LKM1 og læs kommentarerne.

Mere om Python kan I finde her:

<https://docs.python.org/3/>

Generelt så kan I få meget ud af at stille 'Google' spørgsmål på engelsk. Ex.: "How to make loop in python".

Så kommer der ofte gode eksempler I diverse links. Kan dog anbefale et par steder til introduktion:

<https://www.learnpython.org/en/>

<https://wiki.python.org/moin/>

<https://realpython.com/>

Så bare gå til den! 😊 God fornøjelse!

## Trin:

1. Undervisnings stationerne indeholder:
  - a. 2 programmerbare moduler.
    - i. Raspberry PI, som kan sammenlignes med en PC når der er koblet skærm og tastatur til. Man kan koble den til routere, gå på nettet osv.
    - ii. Beocreate board, som er en forstærker med 4 kanaler ud og en digital signalprocessor (DSP) on-board.
  - b. Man kan koble forskellige dimser til - ex. knapper. Vi har koblet en Micor:Bit til , som kan give input til programmerne via seriel kommunikation (koblet til USB)
  - c. Med én ekstra Micro:Bit kan man også sende kommandoer til opstilling på afstand.
  - d. Man kan også koble 1-4 højtalere enheder til så man kan lave forskellige kombinationer. I opstillingen er der to kabinetter med hver sin fuld tone enhed.  
Baggrundsinformation kan findes på nedenstående links, men i dette projekt bør I ikke gå dybt i det. Mere om det i selvstændige forløb.  
<https://www.hifiberry.com/beocreate/beocreate-doc/beocreate-use-sigmastudio/>  
<https://www.raspberrypi.org/>
2. Python er et ud af mange programmeringssprog. Se hjemmesiden for dokumentation og tutorials:  
<https://docs.python.org/3/>  
Download Thonny (IDE med Python integreret) på egen MAC/PC (ver 3.2.1 øverst til højre):  
<https://thonny.org/>
3. "Hello world" - Første lille program, funktioner og betingelser. (som i "M1Sa-Basics.py")
  - a. Biblioteker med funktioner som man kan bygge videre på, så man ikke skal lave alt fra bunden hver gang.  
Installation af Pygame bibliotek på MAC/PC via "Tools/Manage packages" menupunkt i Thonny.  
Dokumentation/manualer for biblioteket findes: <https://www.pygame.org/docs/>
5. Hello world - Nu med Lyd. Reference: <https://www.pygame.org/docs/ref/mixer.html>
  - a. Eksempler fra Github udpakkes til PC/MAC i 'rod direktorie' fra <https://github.com/TPSoundhub/LKM1>
  - b. Afspil lyd i program "Hej med Jer"
  - c. Lav funktion der kan afspille en lyd og dernæst 2 lyde efter hinanden
  - d. Program med navneopråb og baggrund sammen på klassen. (ender som M1Sb-Navne.py)
6. Kør navneopråb fra CLI (kommando linje).
  - a. Køre program fra kommando linje på PI. (i CLI-vinduet)
  - b. Stop det med ctrl-c.
  - c. Indsæt program/script "M1Sd-Navne-xx.py" i filen "rc.local" som er placeret i folderen "Etc"  
Man editere med simpel editor kaldet nano. (Brug "sudo nano rc.local" i etc direktoriet).  
**Husk "&" tegn til sidst i indsat tekst streng.** Det som skal indsættes i bunden af rc.local er:  
"python /home/pi/LKM1/M1Sd-Navne-xx.py &" ( " - tegnene skal ikke med!!)
  - d. Gå i config (med kommandoen "sudo raspi-config") og sæt Pi op til at starte uden GUI med bruger PI uden login
  - e. Genstart PI. Programmet "xxxx.py" kører fra Boot og virker uden at skærm og tastatur skal bruges.
  - f. Gå tilbage og reetabler GUI - ud kommenter/fjern linjen fra rc.local og reboot med GUI

7. Introduktion til Micro:Bit med seriel kode på MAC/PC. Find den serielle kanal.
  - b. Tilslut udleveret Micro:Bit til USB stik på PC/MAC
  - c. Brug M1Sc-VisSerial.py til at finde den port som MB er tilsluttet.
  - d. Brug evt. M1Sc-FindMBserial.py til at vise knapinput, men ellers direkte til 7 og start derfra.
7. Programmer serielt input på PC/MAC – som i M1Sd-Interact.py.
  - a. Opret forbindelse til den serielle kanal
  - b. Lav rutine der læser input fra kanalen og udskriv det der kommer ind. Find positionen med relevant information i strengen og isoler den.
  - c. Lav en funktion der returnerer den relevante information.
  - d. Kobling til en handling – med en lyd i funktion ala den bygget i step 5 men uden at den venter. (Slutter med program som i M1Sd\_Interact.py)
8. Programmer serial input og lyd på PI.
  - a. Gentag på PI
  - b. Samme IDE/samme kode/samme bibliteker. Fokus på forskel i sti/path navngivning.
  - c. Micro:Bit på USB på PI.
  - d. Ende med at køre program ala M1Sd-Interact.py på PI med GUI/Thonny.
8. Headless, og start program fra boot.
  - a. Køre program fra kommando linje på PI, og se at det ikke gør det forventede mht. seriel kommunikation. Brug test udskrift til at se forskel og ret kode til med flag.
  - b. Indsæt program/script "M1Sd-Interact.py" i filen "rc.local" som er placeret i folderen "Etc" Man editere med simpel editor kaldet nano. (Brug "sudo nano rc.local" i etc direktoriet).  
**Husk "&" tegn til sidst i indsat tekst streng.** Det som skal indsættes i bunden af rc.local er:  
"python /home/pi/LKM1/M1Sd-Interact.py &" ( " - tegnene skal ikke med!!)
  - c. Gå ind i config (med kommandoen "sudo raspi-config") og sæt Pi op til at starte uden GUI med bruger PI uden login
  - d. Genstart PI. Programmet "xxxx.py" kører fra Boot og virker uden at skærm og tastatur skal bruges.
  - e. Modsat ud kommenter/fjern linjen fra rc.local og reboot med GUI