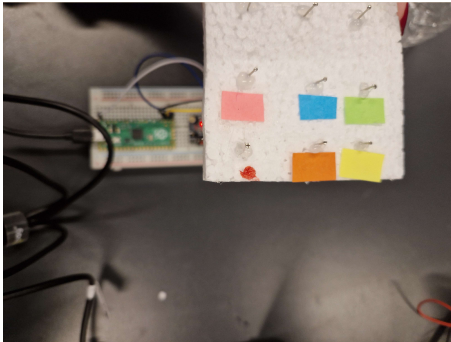


## Projekt 1: Farvespil



```
colors = ["red", "orange", "gul", "grøn", "blå", "pink"]
ch = random.choice(colors)
svaret = "forkert"
point = 0

print(ch)
red = mpr.is_touched(1)
orange = mpr.is_touched(3)
gul = mpr.is_touched(5)
grøn = mpr.is_touched(7)
blå = mpr.is_touched(9)
pink = mpr.is_touched(11)

while True:
    for i in range(12):
        if ch == "red" and mpr.is_touched(1):
            svaret = "korrekt"
            point = point + 1
        elif ch == "orange" and mpr.is_touched(3):
            svaret = "korrekt"
            point = point + 1
        elif ch == "gul" and mpr.is_touched(5):
            svaret = "korrekt"
            point = point + 1
```

vi har lavet et farvespil. computeren vælger en random farve og man skal så trykke på den tilsvarende farvepin på boardet for at få 1 point hvis man rammer den forkerte for man -1 point. det var meningen det skulle være på tid også skulle den vise ens score.

## Projekt 2: Tastaturet



```
while True:
    for i in range(1):
        if mpr.is_touched(11) and mpr.is_touched(10):
            print('a', end='')
            min_tekst = 'a'
            print(min_tekst, end='')
        if mpr.is_touched(11) and mpr.is_touched(9):
            print('b', end='')
        if mpr.is_touched(11) and mpr.is_touched(8):
            print('c', end='')
        if mpr.is_touched(11) and mpr.is_touched(7):
            print('d', end='')
        if mpr.is_touched(11) and mpr.is_touched(6):
            print('e', end='')
        if mpr.is_touched(11) and mpr.is_touched(5):
            print('f', end='')
        if mpr.is_touched(11) and mpr.is_touched(4):
            print('g', end='')
        if mpr.is_touched(11) and mpr.is_touched(3):
            print('h', end='')

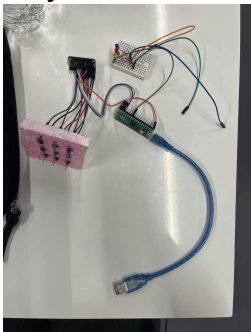
        print(' ', end='')
    if mpr.is_touched(1):
        min_tekst = min_tekst[:-1]
        print(min_tekst)
```

Jeg har lavet et tastatur. Man kan skrive sætninger, lave mellemrum, men kan dog ikke endnu slette bogstaver.

Jeg har prøvet at indsætte en string (min\_tekst += a), men hver gang jeg skriver 'a' bliver der tilføjet et a plus de(t) allerede eksisterende a(er), ellers hvis jeg skriver (som på billedet) min\_tekst = 'a' bliver alle a'er slettet, hvis jeg skriver en række a'er.

Nederste billede er koden af tekst slettelse.

## Projekt 3: Snake



```
alist = [3,4,5,6,7,8,9,10,11]
lst = random.choice(alist)
blst = []

for i in range(3,22):
    if i%10:
        blst.append(i)

lst2 = random.choice(blst)

print(lst)
print(lst2)
print(alist)
print(blst)

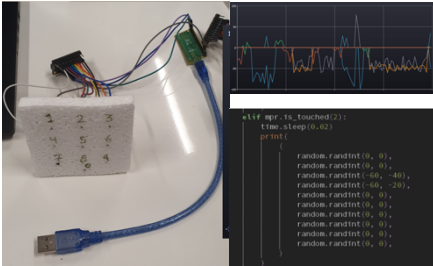
i2c = machine.I2C(0, scl=machine.Pin(17), sda=machine.Pin(21))
mpr = mpr121.MPR121(i2c)
led_pin = Pin(13, Pin.OUT)
point = 0
reset = False

while True:
    for i in range(22):
        led_pin.on()
        time.sleep_ms(200)
        led_pin.off()
        if mpr.is_touched(i):
            print("Key {} pressed".format(i))

        if mpr.is_touched(lst):
            print("Go!")
            point = point + 1
            reset = True
        if mpr.is_touched(lst2):
            print("Go or dead!")
            point = point + 1
            if reset == True:
```

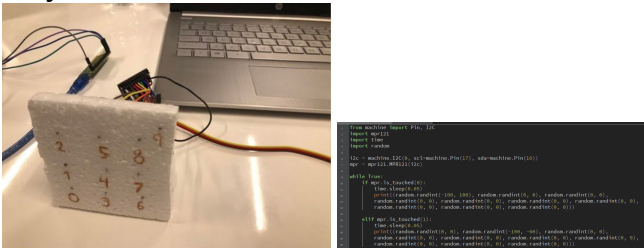
Jeg har lavet en mega bling dims som kan bruges til at spille en halv version af snake, hvor den tilfældigt vælger en god og en dårlig nål - hvis man går hen til den gode får man et point men hvis man vælger den dårlige nål slutter spillet og man har tabt.

## Projekt 4: Grafisk illustrator



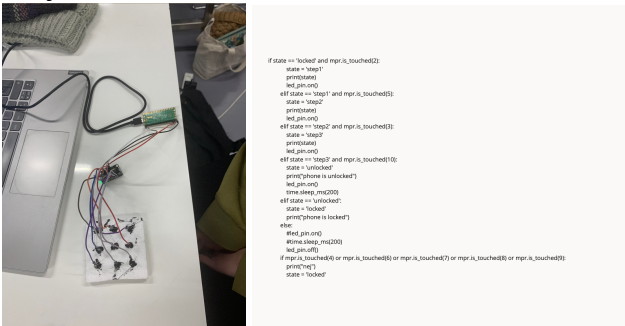
En grafisk illustrator, der plotter en masse grafer alt efter hvilket tal man trykker på

## Projekt 5: Grafter



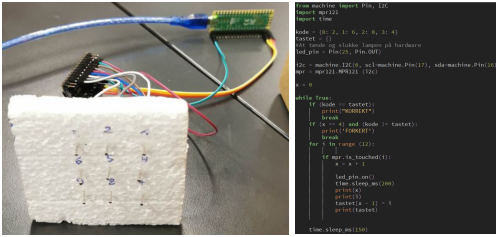
Alt efter hvilket tal man rører ved, vil der komme en tilfældig del af en graf.

## Projekt 6: telefon-kode



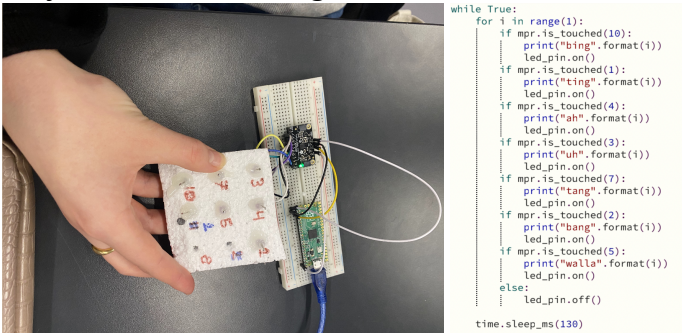
Vi har lavet en kode til en telefon, hvor man skulle skrive den i den rigtige rækkefølge for at den sagde at den var "unlocked", vi fik også gjort sådan at hvis man trykkede forkert så skulle man starte forfra

## Projekt 7: telefon kode



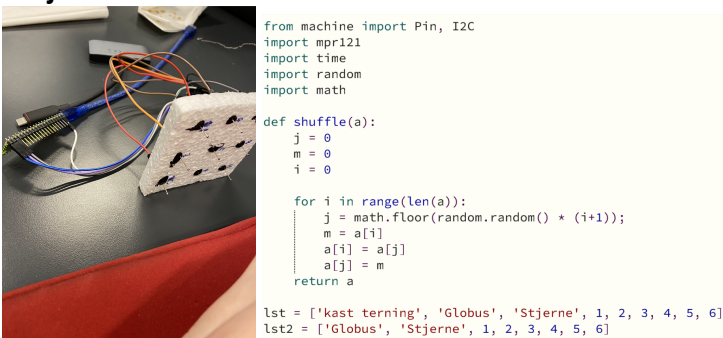
Jeg har lavet et lille kode program ligesom det, der er når man åbner sin telefon. Man kan trykke på de 9 knapper for at trykke på et nummer. Efter 4 tryk vil programmet stoppe og konkludere, om koden er korrekt eller forkert.

## Projekt 8: Cartoons sang



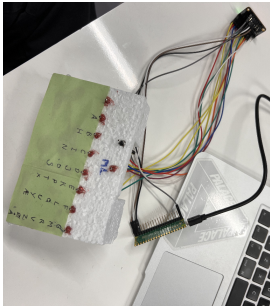
Vi har prøvet at lave "uh ah uh ah ah ting tang walla walla bing bang"-sangen

## Projekt 9: Ludo



Spil der minder om ludo, med terning, globus, stjerne og tal. Man får talene i tilfældig rækkefølge, så man ikke kender udfaldet

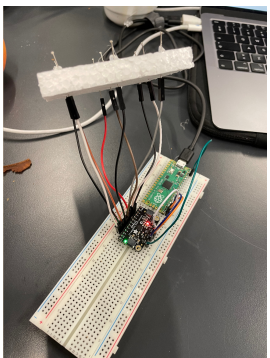
## Projekt 10: Tastatur



```
while True:
    for i in range(1):
        if mpr.is_touched(11) and mpr.is_touched(10):
            print('a',end='')
        if mpr.is_touched(11) and mpr.is_touched(9):
            print('b',end='')
        if mpr.is_touched(11) and mpr.is_touched(8):
            print('c',end='')
        if mpr.is_touched(11) and mpr.is_touched(7):
            print('d',end='')
        if mpr.is_touched(11) and mpr.is_touched(5):
            print('e',end='')
        if mpr.is_touched(11) and mpr.is_touched(4):
            print('f',end='')
        if mpr.is_touched(11) and mpr.is_touched(3):
            print('g',end='')
        if mpr.is_touched(10) and mpr.is_touched(9):
            print('h',end='')
        if mpr.is_touched(10) and mpr.is_touched(8):
            print('i',end='')
        if mpr.is_touched(10) and mpr.is_touched(7):
            print('j',end='')
        if mpr.is_touched(10) and mpr.is_touched(5):
            print('k',end='')
        if mpr.is_touched(10) and mpr.is_touched(4):
```

Jeg har lavet et tastatur, hvor man kan skrive sætninger og lave mellemrum. Man skal bruge 2 fingre for at skrive et bogstav. Hvis bogstavet står på øverste række, skal man placere en finger på den første nål og en anden finger på nålen med det ønskede bogstav på. fx er bogstavet T på nål 4 og 6

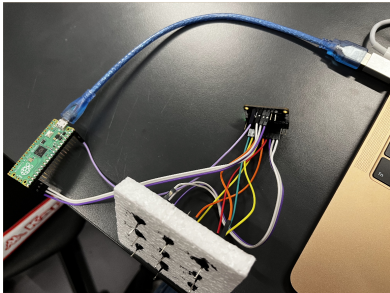
## Projekt 11: Locked



```
32 elif state == 'step1' and mpr.is_touched(5):
33     state = 'step2'
34     led_pin.on()
35     print(state)
36
37 elif state == 'step2' and mpr.is_touched(2):
38     state = 'step3'
39     led_pin.on()
40     print(state)
41
42 elif state == 'step3' and mpr.is_touched(3):
43     state = 'unlocked'
44     led_pin.on()
45     print("phone is unlocked")
46
47 #else:
48 #     state = 'locked'
49 #     # print(state)
50
51 time.sleep_ms(100)
52 led_pin.off()
53
RP2040 REPL
HOW REPLY: CTRL+B TO EXIT
OK
MPY: soft reboot
HOW REPLY: CTRL+B TO EXIT
OK
step1
step2
step3
phone is unlocked
```

Vi prøvede at trykke en 4-cifret kode og så ville vi gerne have at den sagde "locked" indtil koden var rigtig

## Projekt 12: Slange spil



```
import random
import math

def shuffle(a):
    j = 0
    m = 0
    i = 0

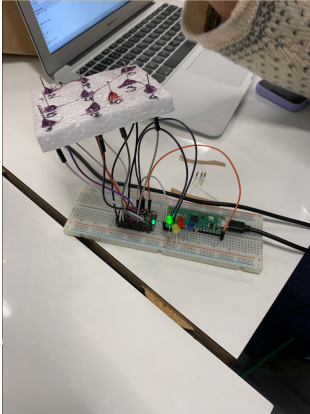
    for i in range(len(a)):
        j = math.floor(random.random() * (i + 1))
        m = a[i]
        a[i] = a[j]
        a[j] = m
    return a

lst = ['slå terning', 'nød', 'op', '2 til venstre', '2 til højre', '5 til venstre', '5 til højre']

while True:
    for i in range(12):
        if mpr.is_touched(3):
            print('træk et kort'.format(i))
            led_pin.on()
            time.sleep_ms(100)
            led_pin.off()
            time.sleep_ms(100)
            led_pin.on()
            newlst = shuffle(lst)
            print(newlst[random.randint(0, len(newlst) - 1)])
```

Jeg har lavet et slange spil, hvor man kan gå op, ned, til højre/venstre, og derudover slå med en terning, og trække et kort, hvor den vælger en random ting/tal fra min liste, eksemplet med at trække et kort kan ses på koden til højre

## Projekt 13: Vende-spil



```
if mpr.is_touched(5) and mpr.is_touched(6):
    print("Hånden du kær")
    led_pin.on()
    time.sleep_ms(2000)
    steat = 0
    led_pin.off()
    return True

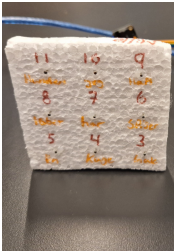
if mpr.is_touched(8) and mpr.is_touched(9):
    print("Hånden du kær")
    led_pin.on()
    time.sleep_ms(2000)
    led_pin.off()
    return True

if mpr.is_touched(7) and mpr.is_touched(10):
    print("Hånden du kær")
    led_pin.on()
    time.sleep_ms(2000)
    led_pin.off()
    steat = 0
    return True

while True:
    for i in range(10):
        if g == "gæt":
            if correct() == True:
                points = points + 1
                print(points)
                g = "gæt"
            if mpr.is_touched(1):
                print("Du låst")
                led_pin.on()
                time.sleep_ms(2000)
                led_pin.off()
                steat = 0
                return False
            if mpr.is_touched(3):
                g = "gæt"
        if g == "gæt":
            if correct() == True:
                points = points + 1
                print(points)
                g = "gæt"
            if mpr.is_touched(1):
                print("Du låst")
                led_pin.on()
                time.sleep_ms(2000)
                led_pin.off()
                print(points)
                g = "gæt"
```

Vi har lavet et "vendespil" hvor du kan spille 2 player. når du gætter rigtig lyser der en grøn lampe. hvis du gætter forkert lyser en rød lampe. Når spillet er slut lyser en blå lampe. hvis du tager for lang tid lyser den gule lampe

## Projekt 14: WordBoard



### Kode

#### Eksempel på individuelle tryk:

```
if mpr.is_touched(0) and i==11:
    print("Hunden")
    led_pin.on()
```

#### Eksempel på dobbelt+ tryk (sætninger):

```
if mpr.is_touched(11) and mpr.is_touched(7) and mpr.is_touched(5) and mpr.is_touched(3):
    print("Hunden har en hale")
    led_pin.on()
```

#### Eksempel på dannelse af datids med har:

```
if mpr.is_touched(11) and mpr.is_touched(7) and mpr.is_touched(8):
    print("Hunden har løbet")
```

Her ses et WordBoard som kan danne sætninger ved at trykke de ønskede ord på samme tid. Ordene kan både trykkes individuelt, men også trykkes sammen til at danne sætninger som: "Hunden har en hale".

Nogle kan trykkes i sammenhæng med "har" for at danne datid.