

Program Miś prawdę Ci powie – do kojarzenia par:)

Do napisania programu -

korzystamy ze strony <https://python.microbit.org/v/3>

gdzie znajduje się edytor kodu do pythona zintegrowany z płytą micorbit

Potrzebujemy dwóch płytek microbit podłączonych do zasilania (baterii).

Poniżej przedstawimy napisany przez nas program na potrzeby konkursu „**Programuję z learn:bit Arcade**”.

Zadaniem programu Miś prawdę Ci powie jest doradzenie „Parom” czy mają przed sobą przyszłość, czy raczej ich związek nie przetrwa próby czasu. Jak Misie (płytki) to robią?

Po pierwsze wyświetlają obrazek i czekają na zmianę pola magnetycznego (przez zbliżenie magnesu). Gdy to się wydarzy, losują cyfrę od 1 do 4. Następnie przesyłają do siebie wylosowaną cyfrę. Dokonują odejmują przestaną cyfrę od wylosowanej przez siebie cyfry. Następnie w zależności od tego czy wylosowane cyfry są identyczne (wynik odejmowania = 0), czy są różne (wynik odejmowania różny od 0) wyświetlają odpowiednią grafikę i grają muzyczkę.

A w programie wygląda to następująco:

Najpierw **importujemy potrzebne moduły:** muzyki, losowania i radia:

```
from microbit import *
```

```
import music
```

```
import random
```

```
import radio
```

Następnie **konfigurujemy kompas a także włączamy radio** i ustawiamy je na odpowiedniej częstotliwości:

```
podstawa = compass.get_z()
```

```
pin1
```

```
radio.on()
```

```
radio.config(group=45)
```

```
scan = i2c.scan()
```

Teraz piszemy podprogram „akcja”

```
def akcja():
```

```
    while True:
```

Piszemy kod, dzięki któremu **na ekranie micorbita pojawi się obrazek celownika**

```
    display.show(Image.TARGET)
```

Teraz piszemy kod odpowiedzialny za uruchomienie programu, **jeżeli zmieni się pole magnetyczne od podstawy** (trzeba to samemu wyczuć ile). Nam wyszło, że 50000 wystarczy

```
if abs (compass.get_z() - podstawa) >= 50000:  
gdy tak się wydarzy, że pole magnetyczne się zmieni microbit zagra melodyjkę
```

```
music.play(music.BA_DING)
```

Następnie piszemy kod, który umożliwi nam **losowanie cyfr z odpowiedniego zakresu**. My wybraliśmy od 1 do 4:

```
random_number = random.randint(1, 4)  
  
display.scroll(random_number)
```

Teraz nasze **płytki wysyłają do siebie wylosowane numery - przez radio**

```
radio.send(str(random_number))  
  
while True:
```

Nasze **płytki prowadzą nasłuch oczekując na wiadomość „message”**

```
message = radio.receive()
```

Niestety **wiadomość nie jest wartością liczbową**, dlatego musimy ją na taką **zamienić za pomocą polecenia int(message)**

```
if message:  
received_num = int(message)
```

Teraz program musi dokonać **działania na wylosowanej wartości i otrzymanej**. Robimy to tak:

```
diff = abs(received_num - random_number)  
  
display.scroll(diff)
```

Jeżeli wynik odejmowania tych wartości równa się 0, **czyli były wylosowane te same liczby, płytka wyświetla serce i wygrywa muzykę weselną**

```
if diff == 0:  
display.show(Image.HEART)  
  
music.play(music.WEDDING)
```

W przeciwnym razie wyświetla „smutną minkę” i gra muzykę, która będzie przygnębiająca, My wybraliśmy tą o nazwie Funeral

```
else:  
display.show(Image.SAD)  
  
music.play(music.FUNERAL)
```

Teraz warto by troszkę ta ikona się powysświetlała a następnie, **czyścimy wyświetlacz płytki**

```
sleep(3000)  
  
display.clear()
```

```
break
```

```
break
```

Jeżeli (w przeciwnym wypadku) **pole magnetyczne się nie zmienia**, chcemy by na wyświetlaczu ciągle pokazywała się grafika „celownik”

else:

```
display.show(Image.TARGET)
```

Teraz **programujemy uruchomienie podprogramu „akcja”** za pomocą naciśnięcia przycisku „a” na płytce

while True:

```
if button_a.is_pressed():
```

```
    akcja()
```

A tak wygląda cały program:

```
from microbit import *
```

```
import music
```

```
import random
```

```
import radio
```

```
podstawa = compass.get_z()
```

```
pin1
```

```
radio.on()
```

```
radio.config(group=45)
```

```
scan = i2c.scan()
```

```
def akcja():
```

```
    while True:
```

```
        display.show(Image.TARGET)
```

```
        if abs (compass.get_z() - podstawa) >= 50000:
```

```
            music.play(music.BA_DING)
```

```
            random_number = random.randint(1, 4)
```

```
            display.scroll(random_number)
```

```
            radio.send(str(random_number))
```

```
while True:
    message = radio.receive()
    if message:
        received_num = int(message)
        diff = abs(received_num - random_number)
        display.scroll(diff)
        if diff == 0:
            display.show(Image.HEART)
            music.play(music.WEDDING)
        else:
            display.show(Image.SAD)
            music.play(music.FUNERAL)
        sleep(3000)
        display.clear()
        break
    break
else:
    display.show(Image.TARGET)
```

```
while True:
    if button_a.is_pressed():
        akcja()
```