

2 PRÁCE S TLAČÍTKY

Co se naučíte

- Ovládat obě programovatelná tlačítka
- Psát programy reagující na stisk tlačítka
- Význam logických spojek and a or

Co budete potřebovat

- PC s nainstalovaným editorem Mu
- Propojovací USB kabel s micro USB koncovkou
- Micro:bit

Časová náročnost

Jedna vyučovací hodina 45 minut

Micro:bit obsahuje celkem tři tlačítka. Tlačítko umístěné na zadní straně mezi vstupy pro USB kabel a napájecí kabel je reset. To nás teď nebude zajímat. Na přední straně jsou umístěna dvě programovatelná tlačítka, A a B, jimž se budeme v této kapitole věnovat.

Začněte jednoduchým příkladem:

```
from microbit import *
while True:
    if button_a.is_pressed():
        display.show(Image.HAPPY)
    if button_b.is_pressed():
        display.show(Image.SAD)
    sleep(100)
    display.clear()
```

Práce s tlačítky je ukázána na řádcích 3 a 5. Jedná se vlastně o dotaz, zda je tlačítko stisknuté. MicroPython má, jak vidíte připraveny dvě proměnné `button_a` a `button_b`. Funkce `button_a.is_pressed()` vrací 1, pokud je tlačítko stisknuté, jinak vrací 0. Existuje ještě funkce `button_a.was_pressed()`, která testuje, zda tlačítko bylo stisknuté od minulé kontroly nebo od zapnutí micro:bitu. Obdobné funkce jsou i pro tlačítko b.

Chcete-li testovat současný stisk obou tlačítek použijte následující programové konstrukci:

```
from microbit import *
while True:
    if (button_a.is_pressed()) and (button_b.is_pressed()):
        display.show(Image.HEART)
        sleep(100)
        display.clear()
```

Mezi oběma testovacími funkcemi na řádce 3 je použita logická spojka `and`, která znamená, že celkově podmínka platí, pouze pokud platí obě dílčí podmínky – jsou stisknuta obě tlačítka současně.

Naopak, pokud testujete, zda je stisklé libovolné tlačítko (A nebo B), použijte následující konstrukci se spojkou `or` (nebo):

```
from microbit import *
while True:
    if ((button_a.is_pressed()) or (button_b.is_pressed())):
        display.show(Image.HEART)
        sleep(100)
        display.clear()
```

Kromě uvedených funkcí `is_pressed` a `was_pressed`, je pro objekty `button_a` a `button_b` definována ještě funkce `get_presses()`. Tato funkce zjistí počet stisknutí tlačítka od posledního testování a nastaví jej na nulu.

Následující příklad vyčká po zapnutí (nebo stisku reset) micro:bitu deset sekund a pak zobrazí počet stisků tlačítka A během této doby (od zapnutí nebo resetu):

```
from microbit import *
sleep(10000)
display.show(button_a.get_presses())
```