

2.看谁按得快

学习目标:

这节课学习使用 Python 语言编程实现在 micro:bit 点阵上实现爱心闪动。

代码与解析:

```
from microbit import *

while True:
    if button_a.is_pressed() is True and button_b.is_pressed() is False:
        sleep(50)
        if button_a.is_pressed() is True and button_b.is_pressed() is False:
            display.show(Image.ARROW_W)
    elif button_a.is_pressed() is False and button_b.is_pressed() is True:
        sleep(50)
        if button_a.is_pressed() is False and button_b.is_pressed() is True:
            display.show(Image.ARROW_E)
    elif button_a.is_pressed() is True and button_b.is_pressed() is True:
        sleep(50)
        if button_a.is_pressed() is True and button_b.is_pressed() is True:
            display.show(Image.HEART)
```

`from microbit import *` 的意思是告诉 MicroPython 我们将要用到 `microbit` 库中的函数, `from` 是从.....地方, `import` 是导入, `*` 在这里的意思是所有。所以 `from microbit import *` 就是从 `microbit` 库中导入所有东西, 在使用 `microbit` 时的每个程序都要导入这个库。

`button_a.is_pressed()`意思是 `microbit` 上的按键 A 被按下的时候, 则返回 `True`, 否则返回 `False`。`button_b.is_pressed()`同理。

`sleep()`的意思是运行到这里的时候暂停指定的时间, 单位为 `ms`。

`display.show(image)`意思在 `micro:bit` 点阵上显示一个内置图像, `display` 也是一个类似于 `microbit` 的库, 只是它被包含在了 `microbit` 里面。`display.show` 的意思就是使用 `display` 这个库里面的 `show()`函数, 这个函数的作用是显示一个内置图像, 而这个内置的图像就是(`Image.HEART`), 一个心形图像, 下面是一份内置图像的清单:

- `Image.HEART`
- `Image.HEART_SMALL`
- `Image.HAPPY`
- `Image.SMILE`
- `Image.SAD`
- `Image.CONFUSED`
- `Image.ANGRY`
- `Image.ASLEEP`

- Image.SURPRISED
- Image.SILLY
- Image.FABULOUS
- Image.MEH
- Image.YES
- Image.NO
- Image.CLOCK12, Image.CLOCK11, Image.CLOCK10, Image.CLOCK9, Image.CLOCK8, Image.CLOCK7, Image.CLOCK6, Image.CLOCK5, Image.CLOCK4, Image.CLOCK3, Image.CLOCK2, Image.CLOCK1
- Image.ARROW_N, Image.ARROW_NE, Image.ARROW_E, Image.ARROW_SE, Image.ARROW_S, Image.ARROW_SW, Image.ARROW_W, Image.ARROW_NW
- Image.TRIANGLE
- Image.TRIANGLE_LEFT
- Image.CHESSBOARD
- Image.DIAMOND
- Image.DIAMOND_SMALL
- Image.SQUARE
- Image.SQUARE_SMALL
- Image.RABBIT
- Image.COW
- Image.MUSIC_CROCHET
- Image.MUSIC_QUAVER
- Image.MUSIC_QUAVERS
- Image.PITCHFORK
- Image.XMAS
- Image.PACMAN
- Image.TARGET
- Image.TSHIRT
- Image.ROLLERSKATE
- Image.DUCK
- Image.HOUSE
- Image.TORTOISE
- Image.BUTTERFLY
- Image.STICKFIGURE
- Image.GHOST

- Image.SWORD
- Image.GIRAFFE
- Image.SKULL
- Image.UMBRELLA
- Image.SNAKE

Image.ALL_CLOCKS

Image.ALL_ARROWS

上面的代码采用两次检测按键是为了防止按键误触导致连续触发按键指定功能，所以在按下后延时一段时间后再次检测按键是否按下。

!特别注意事项:

关于程序编写语法与大小写要求:

1 - 大小写必须正确!

2 - 拼写必须严格正确!

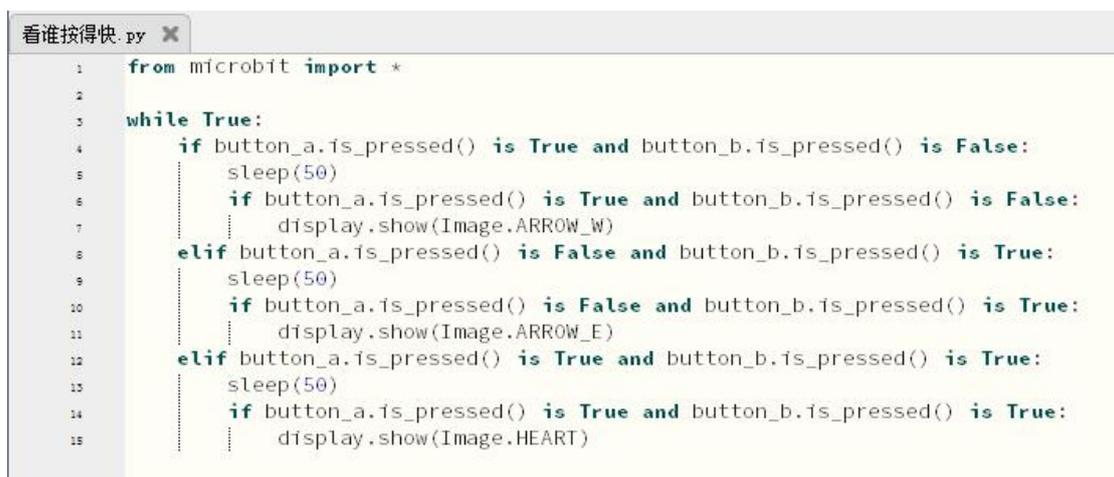
3 - 关键词如 # 与内容之间需要有个空格

4 - 程序以一行空白程序结尾

5 - 程序块体（如 **while** 所包含的程序体是以缩进为标志的），相对于 **C** 语言，**Python** 完全省去了大括号（连同后缀的分号），并使用缩进结构来表示所属关系，只能使用 **Tab** 键（制表键）进行缩进。

编程与下载:

1.打开 Mu 软件，如图 1-1 在编辑窗口输入代码，注意！所有的英文与符号都应该在英文状态下输入，最后一行必须是空格。



```

1  from microbit import *
2
3  while True:
4      if button_a.is_pressed() is True and button_b.is_pressed() is False:
5          sleep(50)
6          if button_a.is_pressed() is True and button_b.is_pressed() is False:
7              display.show(Image.ARROW_W)
8          elif button_a.is_pressed() is False and button_b.is_pressed() is True:
9              sleep(50)
10             if button_a.is_pressed() is False and button_b.is_pressed() is True:
11                 display.show(Image.ARROW_E)
12             elif button_a.is_pressed() is True and button_b.is_pressed() is True:
13                 sleep(50)
14                 if button_a.is_pressed() is True and button_b.is_pressed() is True:
15                     display.show(Image.HEART)

```

图 1-1

2.如图 1-2 所示，点击大拇指图标的检查键检查一下我们的代码是否有错误。如果某一行出现光标或下划线，表示某一行的程序出错，如果没有出现光标或下划线，则表示代码没有错误。

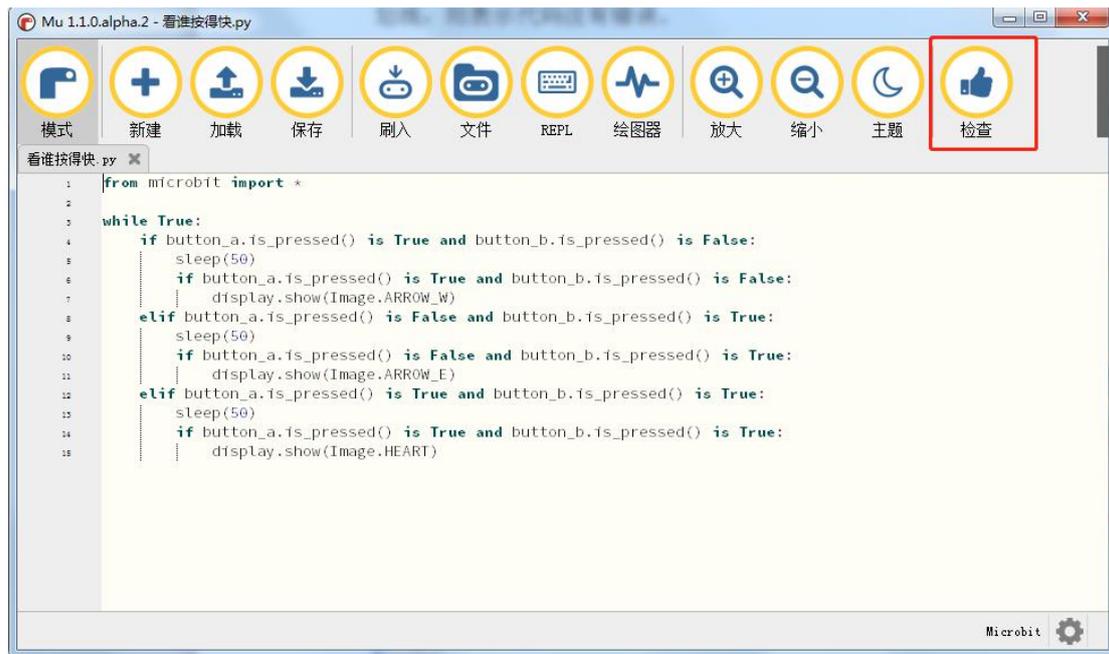


图 1-2

3.将 micro 数据线连接 micro:bit 与电脑，接着如图 1-3 点击刷入按键下载程序至 micro:bit。

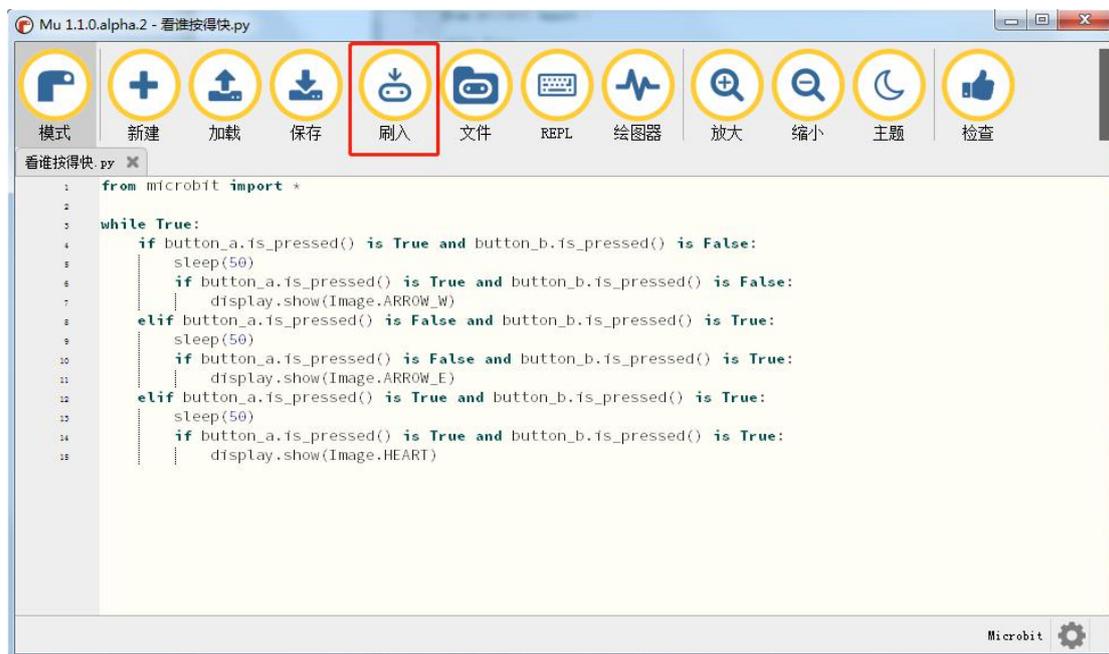


图 1-3

4.下载成功后，大家下载好程序以后，喊上你的小伙伴一起来玩，一个人站在 A 按（àn）键（jiàn）这边，另外一个人站在 B 按（àn）键（jiàn）这边，然后大家一起倒数 3,2,1，一起按下按键，如果 A 按键先被按下的话，点阵上就会

出现一个箭头指向 **A** 按键，如果 **B** 按键先被按下的话，点阵上就会出现一个箭头指向 **B** 按键，如果是同时按下的，点阵上会显示一个爱心哟~