

```

1  # 14: Control de dos LEDs amb intervals de distància
2  # Enunciat: Dissenya un sistema amb dos LEDs que reaccionen a diferents
3  # intervals de distància. El primer LED s'encendrà quan la distància sigui inferior a
4  # 10 cm, indicant proximitat extrema, mentre que el segon LED s'encendrà quan
5  # la distància estigui entre 10 cm i 20 cm, indicant una proximitat moderada.
6  # Utilitza interrupcions per activar i desactivar els LEDs segons aquests intervals
7  # de distància.
8
9  import RPi.GPIO as GPIO
10 import time
11 TRIG = 23
12 ECHO = 24
13 RED_LED_PIN = 17
14 YELLOW_LED_PIN = 27
15 GPIO.setmode(GPIO.BCM)
16 GPIO.setup(TRIG, GPIO.OUT)
17 GPIO.setup(ECHO, GPIO.IN)
18 GPIO.setup(RED_LED_PIN, GPIO.OUT)
19 GPIO.setup(YELLOW_LED_PIN, GPIO.OUT)
20 def mesurar_distancia():
21     GPIO.output(TRIG, True)
22     time.sleep(0.00001)
23     GPIO.output(TRIG, False)
24
25     while GPIO.input(ECHO) == 0:
26         start_time = time.time()
27     while GPIO.input(ECHO) == 1:
28         end_time = time.time()
29
30     duration = end_time - start_time
31     distance = (duration * 34300) / 2
32     return distance
33
34 def controlar_leds(channel):
35     distancia = mesurar_distancia()
36     if distancia < 10:
37         GPIO.output(RED_LED_PIN, True)
38         GPIO.output(YELLOW_LED_PIN, False)
39     elif 10 <= distancia <= 20:
40         GPIO.output(RED_LED_PIN, False)
41         GPIO.output(YELLOW_LED_PIN, True)
42     else:
43         GPIO.output(RED_LED_PIN, False)
44         GPIO.output(YELLOW_LED_PIN, False)
45
46 GPIO.add_event_detect(ECHO, GPIO.BOTH, callback=controlar_leds)
47
48 try:
49     while True:
50         time.sleep(1)
51 except KeyboardInterrupt:
52     print("Programa finalitzat")
53 finally:
54     GPIO.cleanup()

```