```
import time
from adafruit_crickit import crickit

ss = crickit.seesaw

L1 = crickit.SIGNAL1   # limit switch L1
L2 = crickit.SIGNAL8   # limit switch L2

ss.pin_mode(L1, ss.INPUT_PULLDOWN)
ss.pin_mode(L2, ss.INPUT_PULLDOWN)

print("Reading limit switches")

while True:
   print("L1 :", ss.digital_read(L1), "   L2 :", ss.digital_read(L2))
   time.sleep(0.2)
```

Code explanation

import time
This loads the time library so the program can pause for short periods using sleep.

from adafruit_crickit import crickit
This imports the Crickit library so we can access its signal pins and sensors.

ss = crickit.seesaw
This creates a shortcut to the seesaw interface, which is used to read digital and analog pins on Crickit.

L1 = crickit.SIGNAL1
This sets SIGNAL1 as the first limit switch, which is used to detect when the mechanism is fully open.

L2 = crickit.SIGNAL8
This sets SIGNAL8 as the second limit switch, which is used to detect when the mechanism is fully closed.

ss.pin_mode(L1, ss.INPUT_PULLDOWN)
This configures the first limit switch pin as an input with a pull down resistor so it normally reads 0 and becomes 1 when pressed.

ss.pin_mode(L2, ss.INPUT_PULLDOWN)
This does the same setup for the second limit switch.

print("Reading limit switches")
This prints a message so you know the program has started.

while True:
This creates an infinite loop so the program keeps running and checking the switches.

print("L1 (Open):", ss.digital_read(L1), " L2 (Close):", ss.digital_read(L2))
This reads both limit switches and prints their current states.
A value of 1 means the switch is pressed, and 0 means it is not pressed.

time.sleep(0.2)
This waits for a short time before reading again, so the output is easy to read and does not scroll too fast.