

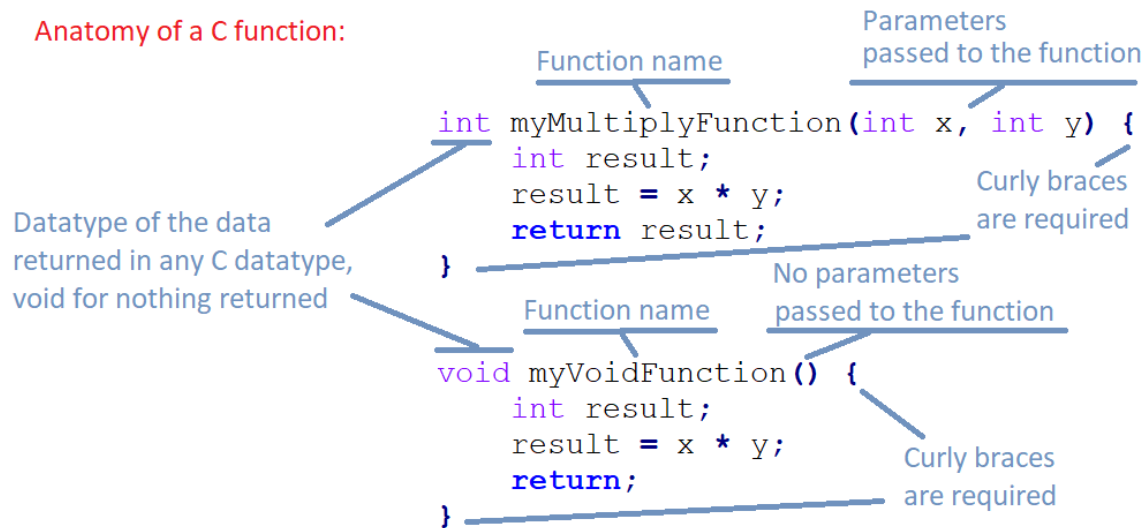


# Microcontrollers & Applications

Lecture 3.2: User-Defined Functions & Variable Scope

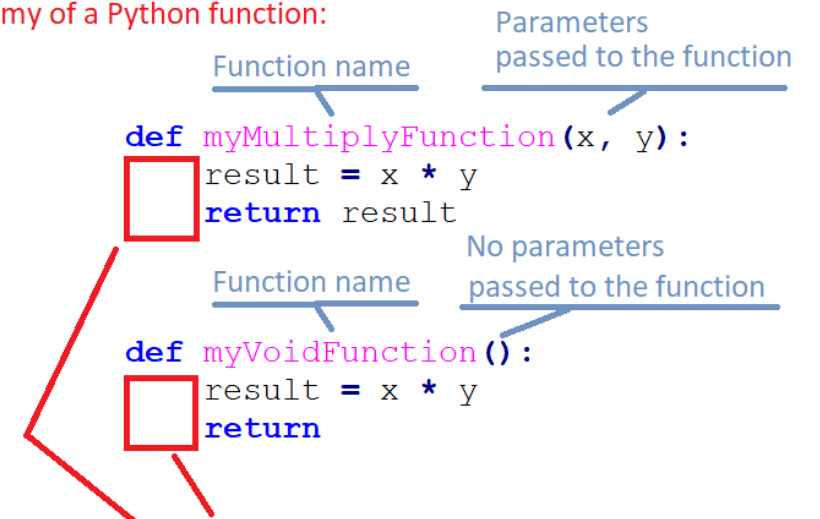
# Function Definition

## Anatomy of a C function:



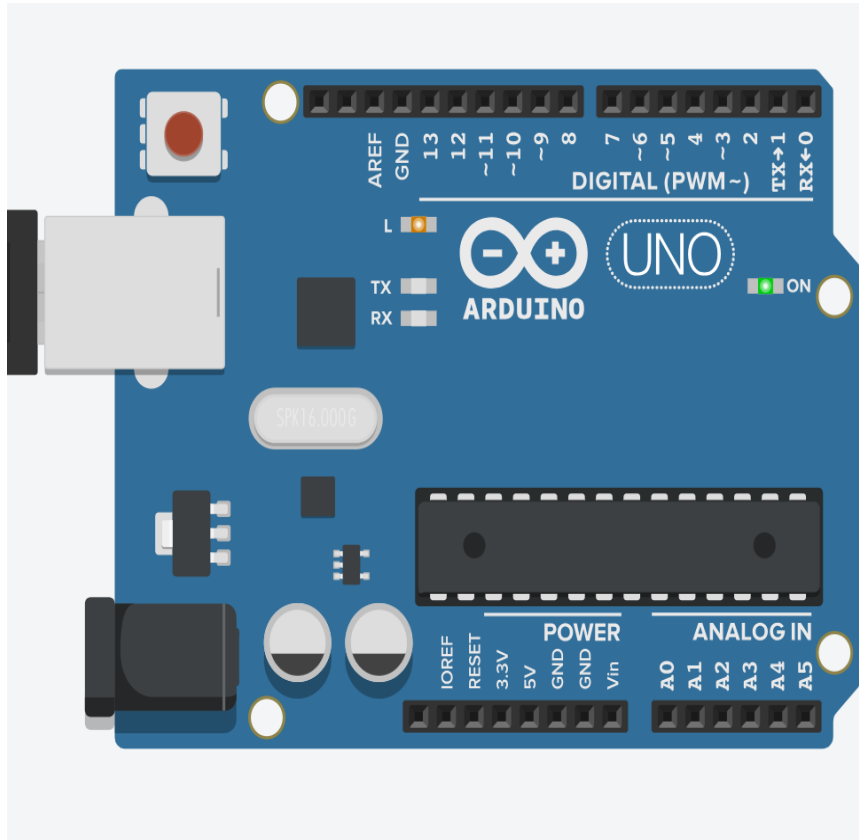
return statements, datatype of returned value must match the datatype declaration of the function

## Anatomy of a Python function:



Giving the same number of extra spaces from the left is necessary to indicate which commands belong to the function

# Function Definition in Arduino (1)



```
1 (Arduino Uno R3)
1 #define ONBOARD_LED 13
2 #define DELAY_TIME 1000
3
4 void setup()
5 {
6   pinMode(ONBOARD_LED, OUTPUT);
7   Serial.begin(9600);
8   Serial.println("LED functions under testing");
9 }
10
11 void loop()
12 {
13   led_on();
14   led_off();
15 }
16
17 void led_on()
18 {
19   digitalWrite(ONBOARD_LED, HIGH);
20   Serial.println("LED is lightening");
21   delay(DELAY_TIME);
22 }
23
24 void led_off()
25 {
26   digitalWrite(ONBOARD_LED, LOW);
27   Serial.println("LED is NOT lightening");
28   delay(DELAY_TIME);
29 }
```

Serial Monitor

```
LED is NOT lightening
LED is lightening
LED is NOT lightening
LED is lightening
LED is NOT lightening
LED is lightening
LED is NOT lightening
LED is lightening
LED is NOT lightening
LED is lightening
LED is NOT lightening
LED is lightening
```

```
#define ONBOARD_LED 13
```

```
#define DELAY_TIME 1000
```

```
void setup()
```

```
  pinMode(ONBOARD_LED,
  OUTPUT);
```

```
  Serial.begin(9600);
```

```
  Serial.println("LED functions
  under testing");
```

```
void loop()
```

```
  led_on();
```

```
  led_off();
```

```
}
```

```
void led_on()
```

```
{
```

```
  digitalWrite(ONBOARD_LED,
  HIGH);
```

```
  Serial.println("LED is lightening");
```

```
  delay(DELAY_TIME);
```

```
}
```

```
void led_off()
```

```
{
```

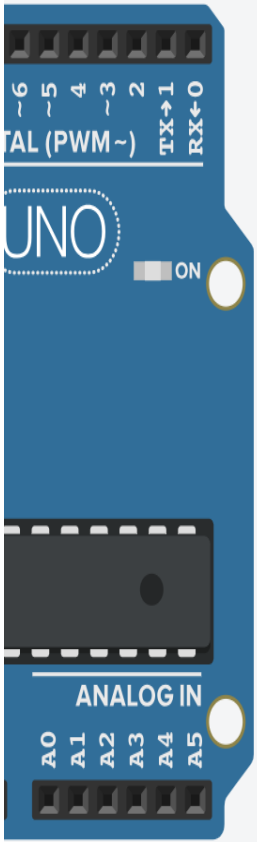
```
  digitalWrite(ONBOARD_LED,
  LOW);
```

```
  Serial.println("LED is NOT
  lightening");
```

```
  delay(DELAY_TIME);
```

```
}
```

# Function Definition in Arduino (2): variable scope



```
1 unsigned int ONBOARD_LED = 13;
2 unsigned long DELAY_TIME = 1000;
3
4 void setup()
5 {
6   unsigned long BPS = 9600;
7   pinMode(ONBOARD_LED, OUTPUT);
8   Serial.begin(BPS);
9   Serial.println("LED functions under testing");
10 }
11
12 void loop()
13 {
14   unsigned long MY_TIME = 1000;
15   led_on();
16   led_off();
17   led_toggle();
18 }
19
20 void led_on()
21 {
22   unsigned long DELAY_TIME = 500;
23   digitalWrite(ONBOARD_LED, HIGH);
24   Serial.println("LED is lightening");
25   delay(DELAY_TIME);
26 }
27
28 void led_off()
29 {
30   digitalWrite(ONBOARD_LED, LOW);
31   Serial.println("LED is NOT lightening");
32   delay(DELAY_TIME);
33 }
34
35 void led_toggle()
36 {
37   Serial.println("LED is TOGGING");
38   delay(MY_TIME);
39 }
```

Sorry, it seems like your code has some errors.

Close

```
In function 'void led_toggle()':
38:9: error: 'MY_TIME' was not declared in this scope
38:9: note: suggested alternative: 'DELAY_TIME'
```

```
unsigned int ONBOARD_LED = 13;
```

```
unsigned long DELAY_TIME =
1000;
```

```
void setup()
```

```
{
  unsigned long BPS = 9600;
  pinMode(ONBOARD_LED,
OUTPUT);
```

```
Serial.begin(BPS);
Serial.println("LED functions
under testing");
```

```
}
```

```
void loop()
```

```
{
  unsigned long MY_TIME = 1000;
```

```
led_on();
```

```
led_off();
```

```
led_toggle();
```

```
}
```

```
void led_on()
```

```
{
  unsigned long DELAY_TIME =
500;
```

```
digitalWrite(ONBOARD_LED,
HIGH);
```

```
Serial.println("LED is
lightening");
```

```
delay(DELAY_TIME);
```

```
}
```

```
void led_off()
```

```
{
```

```
digitalWrite(ONBOARD_LED,
LOW);
```

```
Serial.println("LED is NOT
lightening");
```

```
delay(DELAY_TIME);
```

```
}
```

```
void led_toggle()
```

```
{
```

```
DELAY_TIME = 2000;
```

```
Serial.println("LED is
TOGGING");
```

```
delay(MY_TIME);
```

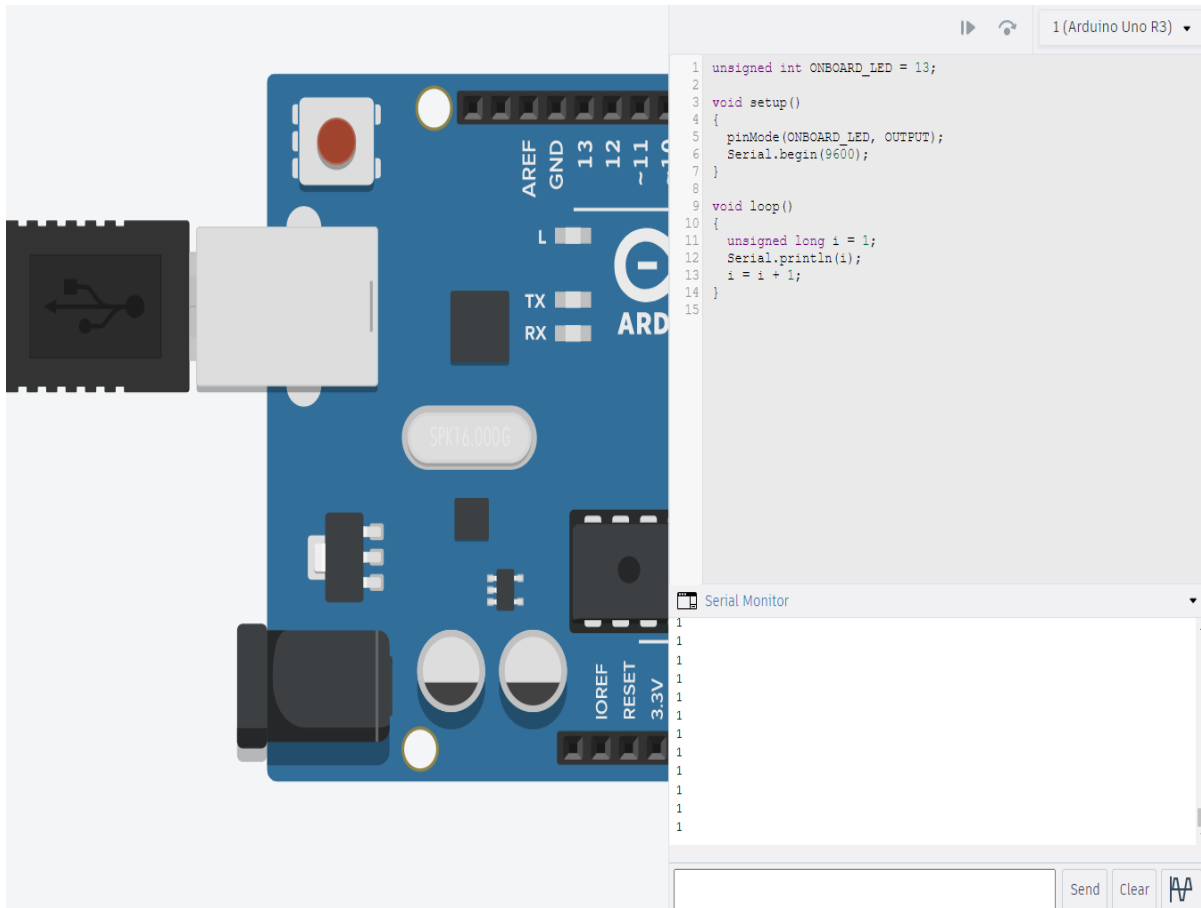
```
}
```

```
In function 'void led_toggle()':
38:9: error: 'MY_TIME' was not declared
in this scope
38:9: note: suggested alternative:
'DELAY_TIME'
```

## Function Definition in Arduino (3): global vs local variables

- A variable must be defined before its first use.
- A variable must be defined in the same code block (limited by curly braces) or in the upper code block.
  - `ONBOARD_LED` is defined at the top of the program, so you can use it everywhere.
  - `MY_TIME` is defined under the `loop()` function. Since `led_toggle()` function is not a sub-block of the `loop()` function block, it generates an error message.
  - `DELAY_TIME` is defined both at the top of the program and the `led_on()` function block. The value of the `DELAY_TIME` is 500 in the `led_on()` code block, 1000 in other code blocks.
  - `DELAY_TIME` is changed in the `led_toggle()` function block, so that `DELAY_TIME` will be 2000 in all code blocks (other than `led_on`) after calling `led_toggle()` function.
- If a variable is defined in both in code block and its upper block, the definition of the code block itself will be used.

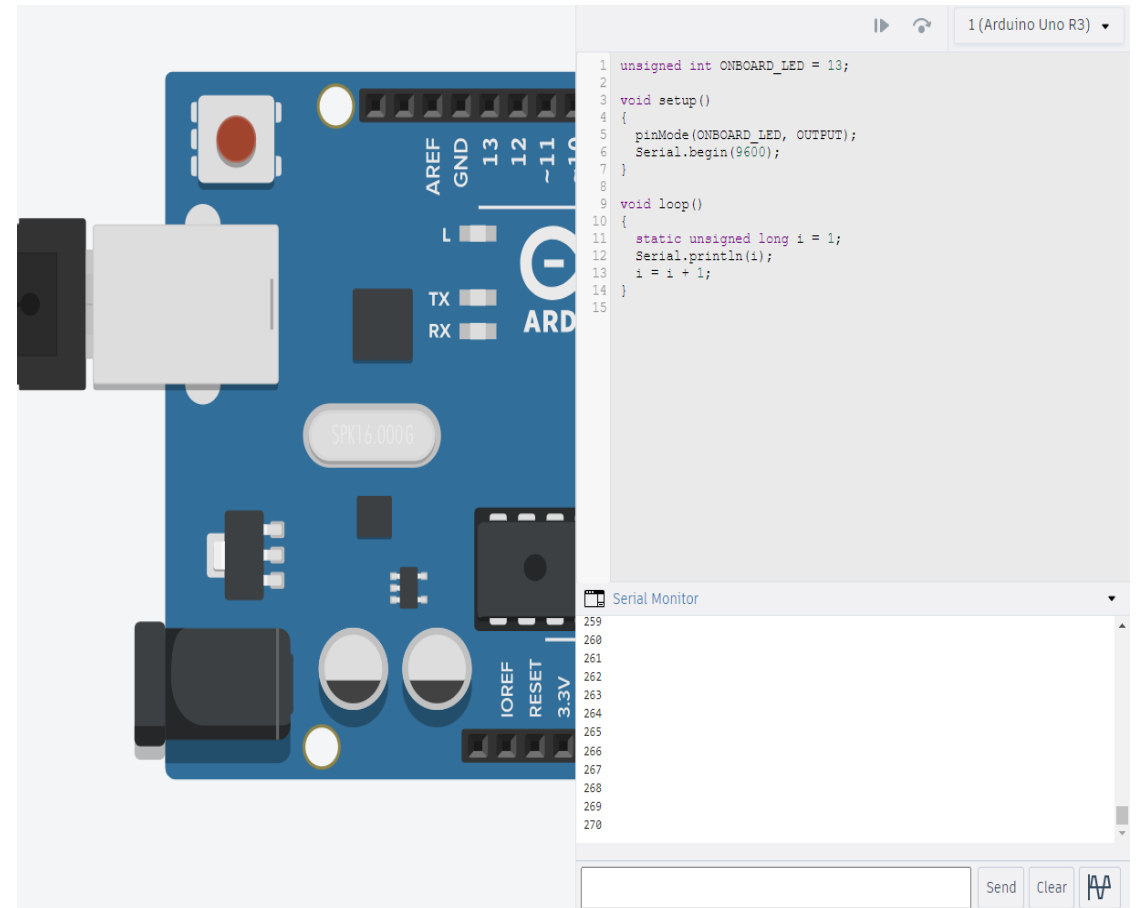
# Function Definition in Arduino (4): static variables



The screenshot shows the Arduino IDE interface. On the left is a 3D model of an Arduino Uno R3 board. On the right, the code editor displays the following sketch:

```
1 unsigned int ONBOARD_LED = 13;
2
3 void setup()
4 {
5   pinMode(ONBOARD_LED, OUTPUT);
6   Serial.begin(9600);
7 }
8
9 void loop()
10 {
11   unsigned long i = 1;
12   Serial.println(i);
13   i = i + 1;
14 }
15
```

Below the code editor is the Serial Monitor window, which is currently empty, indicating that the sketch has not yet been executed.

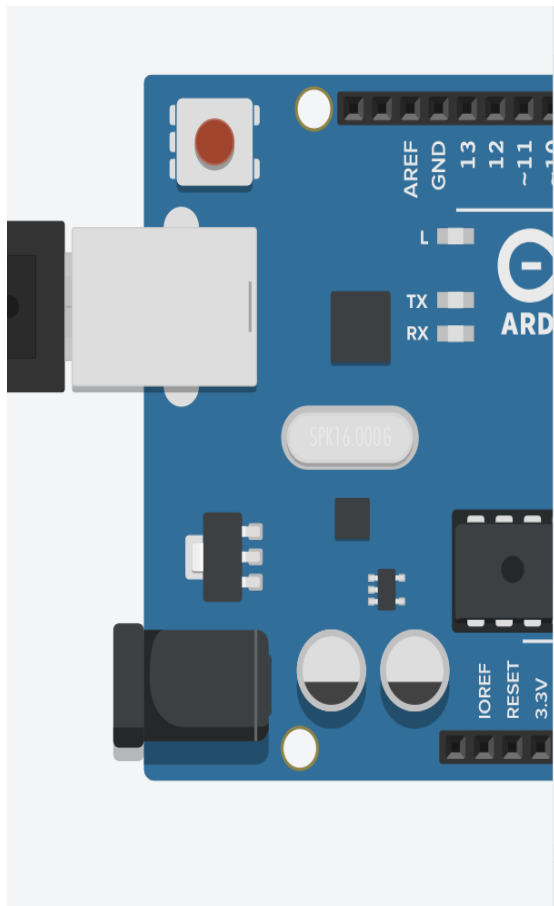


The screenshot shows the same Arduino IDE interface as the previous image, but with a modification to the code. The variable `i` is now declared as `static unsigned long` in the `loop()` function. The Serial Monitor window now displays the output of the sketch:

```
1
259
260
261
262
263
264
265
266
267
268
269
270
```

The output consists of a series of '1's, one on each line, demonstrating that the variable `i` retains its value across function calls due to its static nature.

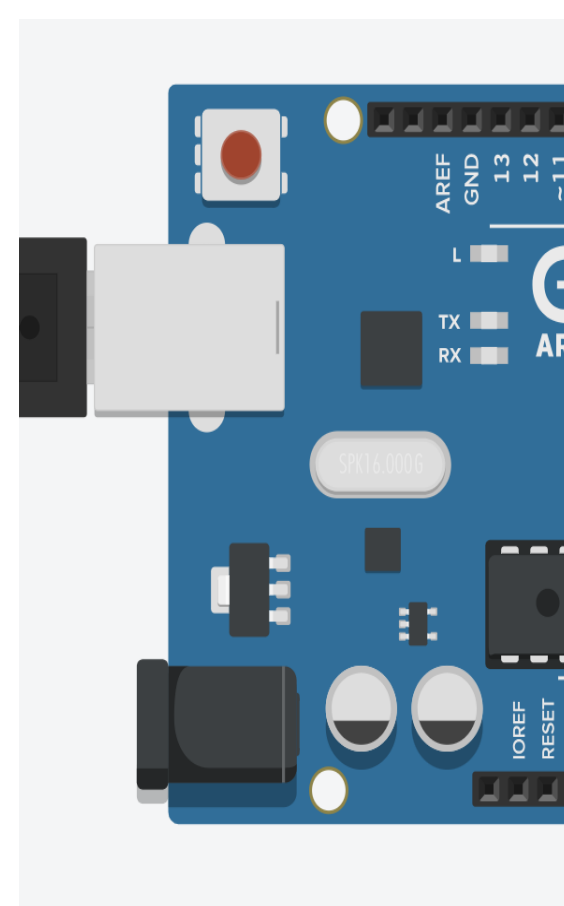
# Function Definition in Arduino (4): argument passing byval vs byref



Arduino Uno R3 board with a code editor window overlaid. The code editor shows the following code:

```
1 unsigned int ONBOARD_LED = 13;
2
3 void setup()
4 {
5   pinMode(ONBOARD_LED, OUTPUT);
6   Serial.begin(9600);
7 }
8
9 void loop()
10 {
11   static unsigned long i = 1;
12   Serial.println(i);
13   increment_one(i);
14 }
15
16 void increment_one(unsigned long x)
17 {
18   x = x + 1;
19 }
20
```

Serial Monitor window is open and empty.

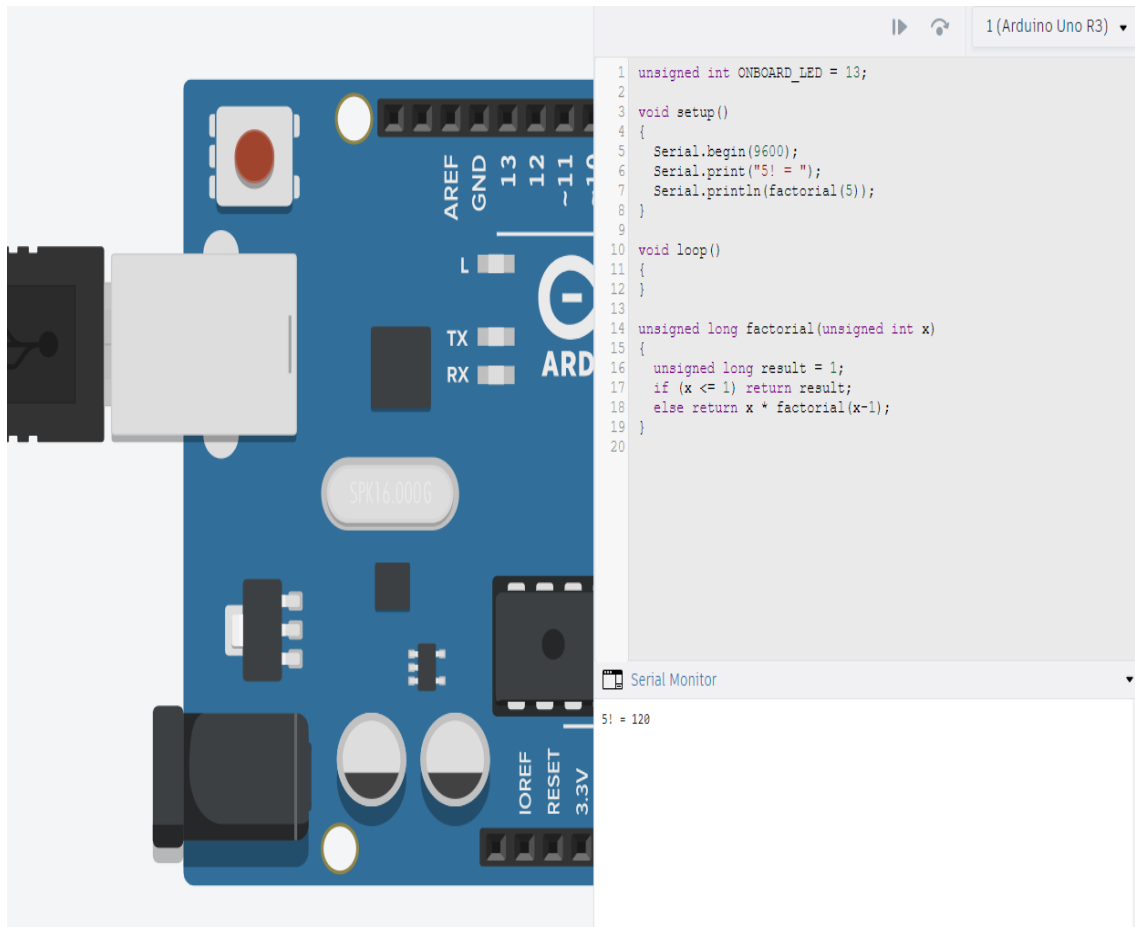


Arduino Uno R3 board with a code editor window overlaid. The code editor shows the following code:

```
1 unsigned int ONBOARD_LED = 13;
2
3 void setup()
4 {
5   pinMode(ONBOARD_LED, OUTPUT);
6   Serial.begin(9600);
7 }
8
9 void loop()
10 {
11   static unsigned long i = 1;
12   Serial.println(i);
13   increment_one(&i);
14 }
15
16 void increment_one(unsigned long *x)
17 {
18   *x = *x + 1;
19 }
20
```

Serial Monitor window is open and empty.

# Function Definition in Arduino (5): recursive functions



```
void setup() {
  Serial.begin(9600);
  Serial.print("5! = ");
  Serial.println(factorial(5));
  // Serial.println("5! = " + factorial(5));
}
```

```
void loop() {
}
```

```
unsigned long factorial(unsigned int x) {
  unsigned long result = 1;
  if (x <= 1) return result;
  else return x * factorial(x-1);
}
```



## Function Definition in Python:

- A variable must have a value before its first read.
- All variables become global variable and are available to all code blocks.



Thanks for  
listening 😊

YALÇIN İŞLER

Assoc. Prof.