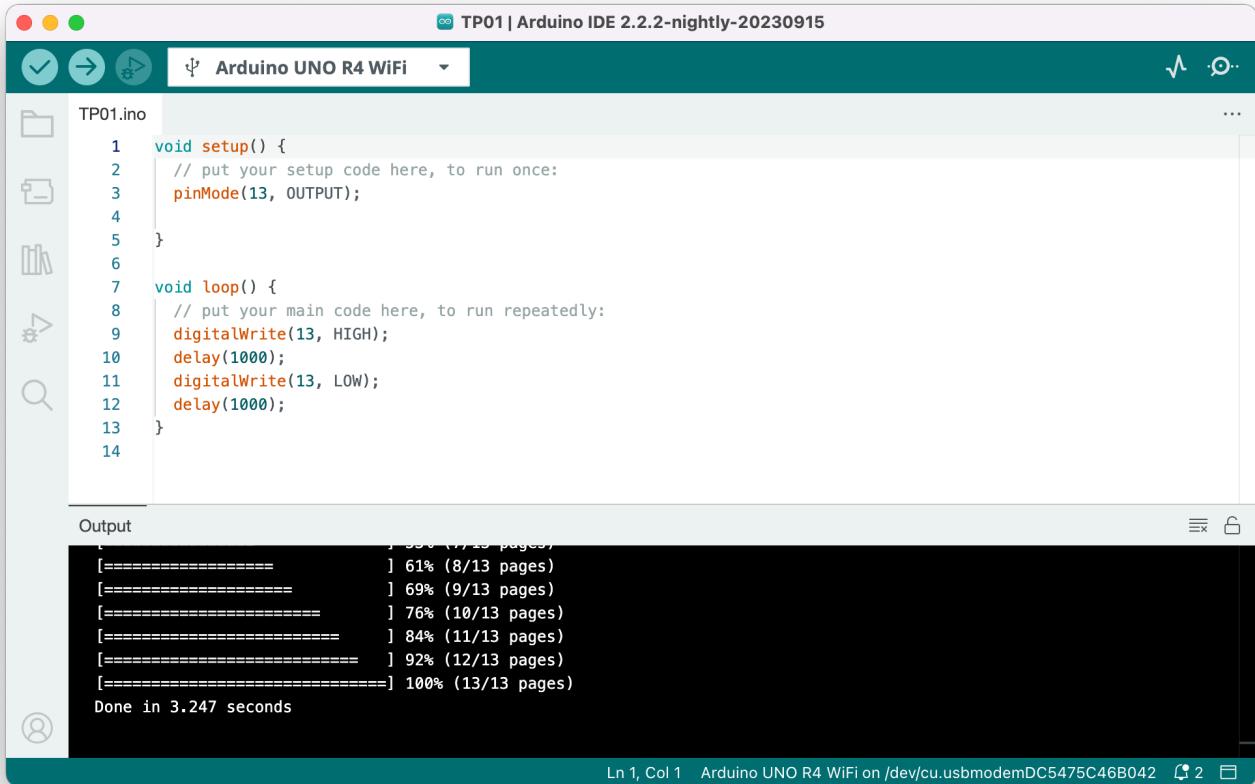


Programmation Multitâches – TP01

Exercice 01



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbars:** Standard Arduino IDE toolbars for file, edit, and build.
- Code Editor:** The file "TP01.ino" contains the following code:

```
1 void setup() {
2     // put your setup code here, to run once:
3     pinMode(13, OUTPUT);
4
5 }
6
7 void loop() {
8     // put your main code here, to run repeatedly:
9     digitalWrite(13, HIGH);
10    delay(1000);
11    digitalWrite(13, LOW);
12    delay(1000);
13 }
```
- Output Window:** Shows the serial monitor output:

```
[=====] 55% (7/13 pages)
[=====] 61% (8/13 pages)
[=====] 69% (9/13 pages)
[=====] 76% (10/13 pages)
[=====] 84% (11/13 pages)
[=====] 92% (12/13 pages)
[=====] 100% (13/13 pages)
Done in 3.247 seconds
```
- Status Bar:** Ln 1, Col 1 Arduino UNO R4 WiFi on /dev/cu.usbmodemDC5475C46B042 2

Voici le sens des différentes lignes :

La fonction `loop()` va allumer la LED de test D13 pendant une seconde, puis l'éteindre pendant une seconde. Cette fonction est appelée de façon infinie faisant donc clignoter la LED de façon infinie également.

La fonction `pinMode(13, OUTPUT)`, dans la fonction `setup()`, va avoir la charge d'initialiser la broche 13 en sortie.

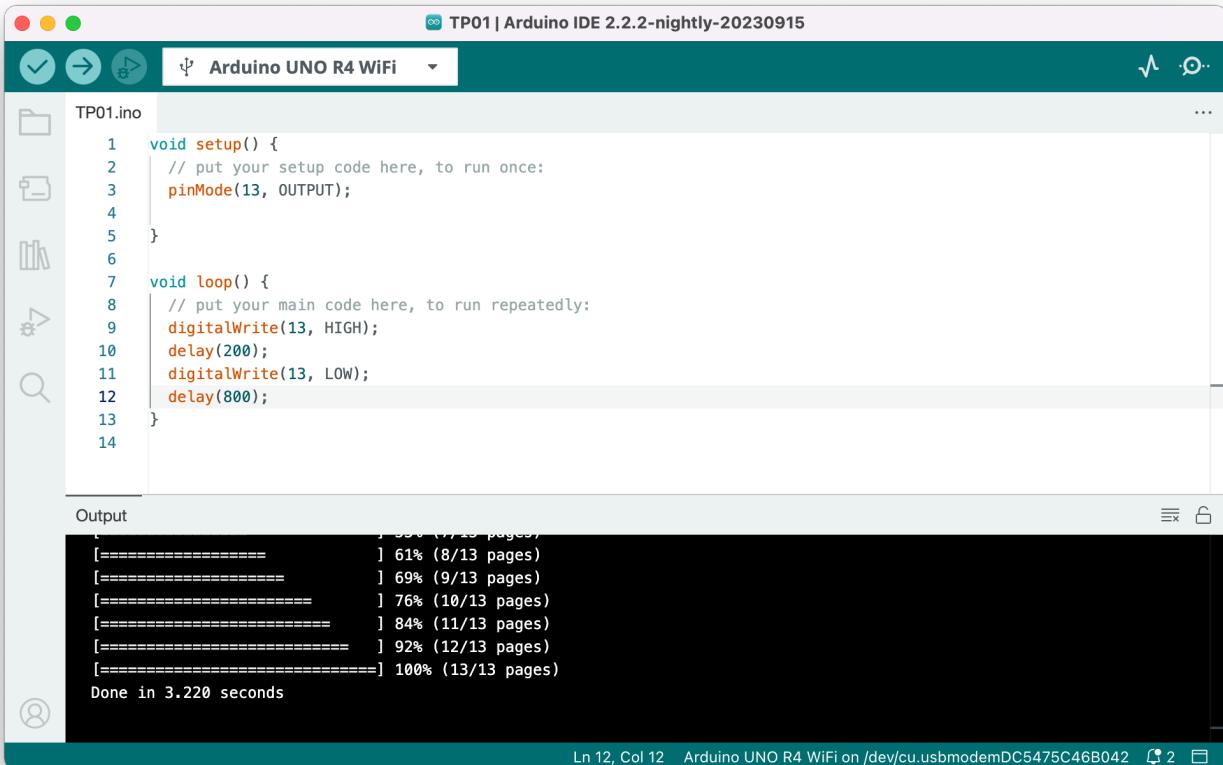
Exercice 02

Question 1

Avec le code montré ci-dessus, la LED clignote déjà à une fréquence de 1Hz.

Question 2

Voici le code pour avoir un clignotement avec un rapport cyclique de 20%.



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbars:** Standard Arduino IDE toolbars for file operations (New, Open, Save, Print, etc.) and project management.
- Left Sidebar:** Project browser showing "TP01.ino".
- Code Editor:** Contains the following Arduino sketch:

```
TP01.ino
1 void setup() {
2     // put your setup code here, to run once:
3     pinMode(13, OUTPUT);
4 }
5
6
7 void loop() {
8     // put your main code here, to run repeatedly:
9     digitalWrite(13, HIGH);
10    delay(200);
11    digitalWrite(13, LOW);
12    delay(800);
13 }
14 }
```
- Output Window:** Shows the compilation progress and completion message:

```
[=====] 55% (7/13 pages)
[=====] 61% (8/13 pages)
[=====] 69% (9/13 pages)
[=====] 76% (10/13 pages)
[=====] 84% (11/13 pages)
[=====] 92% (12/13 pages)
[=====] 100% (13/13 pages)
Done in 3.220 seconds
```
- Bottom Status Bar:** Displays "Ln 12, Col 12" and "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

Exercice 03

Question 01 et 02

Les explications sont dans les commentaires.



The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The left sidebar shows a file named "TP01.ino". The code in the editor is:

```
TP01.ino
1 void setup() {
2     pinMode(13, OUTPUT);
3     Serial.begin(9600);
4 }
5
6 void loop() {
7     digitalWrite(13, HIGH);
8     Serial.print("H"); // Print the string without \n at the end
9     delay(200);
10    digitalWrite(13, LOW);
11    Serial.println("L"); // Add a \n at the end of the line
12    delay(800);
13 }
```

The "Serial Monitor" tab is selected, showing the output:

```
Output Serial Monitor ×
Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodemDC5475C46B042')
New Line 9600 baud
HL
HL
HL
HL
HL
HL
```

The status bar at the bottom indicates "Ln 3, Col 1 Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042" and "2" open files.

Question 03

Il semblerait qu'il faille appuyer sur reset pour pouvoir recharger le contenu de la fonction setup(). Une fois cette tâche effectuée, on obtient le résultat suivant :



The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The left sidebar shows a file named "TP01.ino". The code in the editor is:

```
TP01.ino
1 void setup() {
2     pinMode(13, OUTPUT);
3     Serial.begin(9600);
4     Serial.println("Bonjour!");
5 }
6
7 void loop() {
8     digitalWrite(13, HIGH);
9     delay(200);
10    digitalWrite(13, LOW);
11    delay(800);
12 }
```

The "Serial Monitor" tab is selected, showing the output:

```
Output Serial Monitor ×
Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodemDC5475C46B042')
No Line Ending 9600 baud
09:04:20.627 -> Bonjour!
09:04:30.871 -> Bonjour!
```

The status bar at the bottom indicates "Ln 4, Col 26 Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042" and "2" open files.

Question 04

Voici le code de la question 04. La concaténation de deux types n'étant pas possible sans `cast` en C, on partira du principe qu'un nouveau `print()` sera plus optimisé.



The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The left sidebar shows a file tree with "TP01.ino" selected. The main code editor window contains the following code:

```
TP01.ino
1 int counter = 0;
2
3 void loop() {
4     counter += 1;
5
6     digitalWrite(13, HIGH);
7     Serial.print("H"); // Print the string without \n at the end
8     delay(200);
9     digitalWrite(13, LOW);
10    Serial.print("L");
11    Serial.print(" Nombre de boucle(s) = ");
12    Serial.println(counter);
13
14    delay(800);
15
16}
```

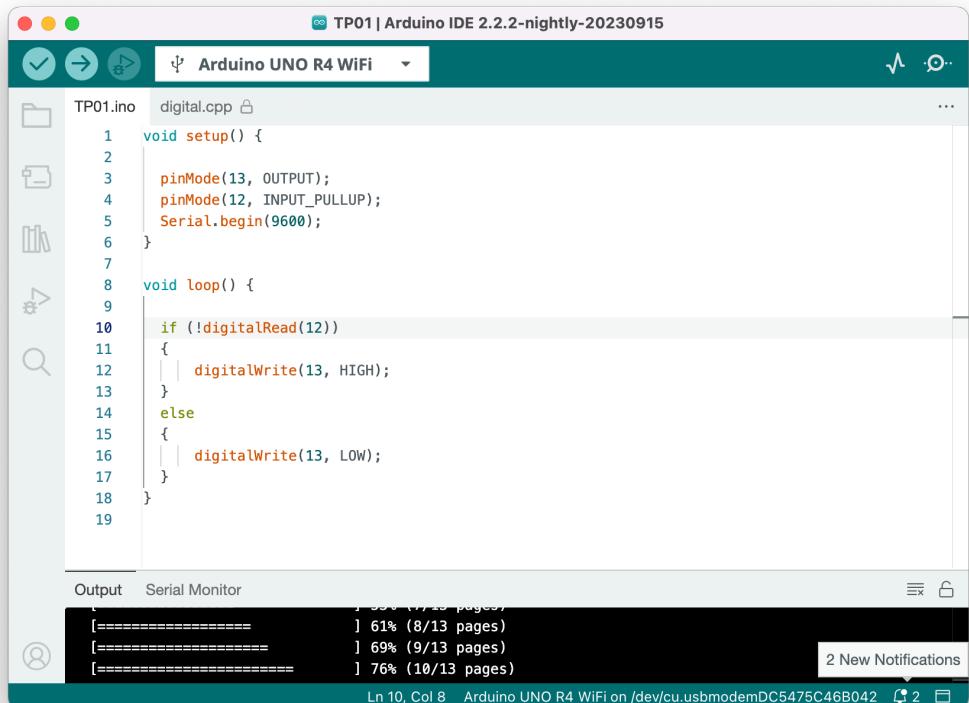
The "Output" tab is active, showing the serial monitor output:

```
Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodemDC5475C46B042')
No Line Ending 9600 baud
09:08:25.564 -> HLNombre de boucle(s) = 6
09:08:26.620 -> HLNombre de boucle(s) = 7
09:08:27.687 -> HLNombre de boucle(s) = 8
09:08:28.782 -> HLNombre de boucle(s) = 9
09:08:29.862 -> HLNombre de boucle(s) = 10
09:08:30.928 -> HLNombre de boucle(s) = 11
```

The status bar at the bottom indicates "Ln 18, Col 27" and "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

Exercice 04

Voici le code permettant d'allumer la LED lors de l'appui sur le bouton poussoir.



The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The left sidebar shows a file tree with "TP01.ino" selected. The main code editor window contains the following code:

```
TP01.ino
1 void setup() {
2
3     pinMode(13, OUTPUT);
4     pinMode(12, INPUT_PULLUP);
5     Serial.begin(9600);
6 }
7
8 void loop() {
9
10    if (!digitalRead(12))
11    {
12        digitalWrite(13, HIGH);
13    }
14    else
15    {
16        digitalWrite(13, LOW);
17    }
18 }
```

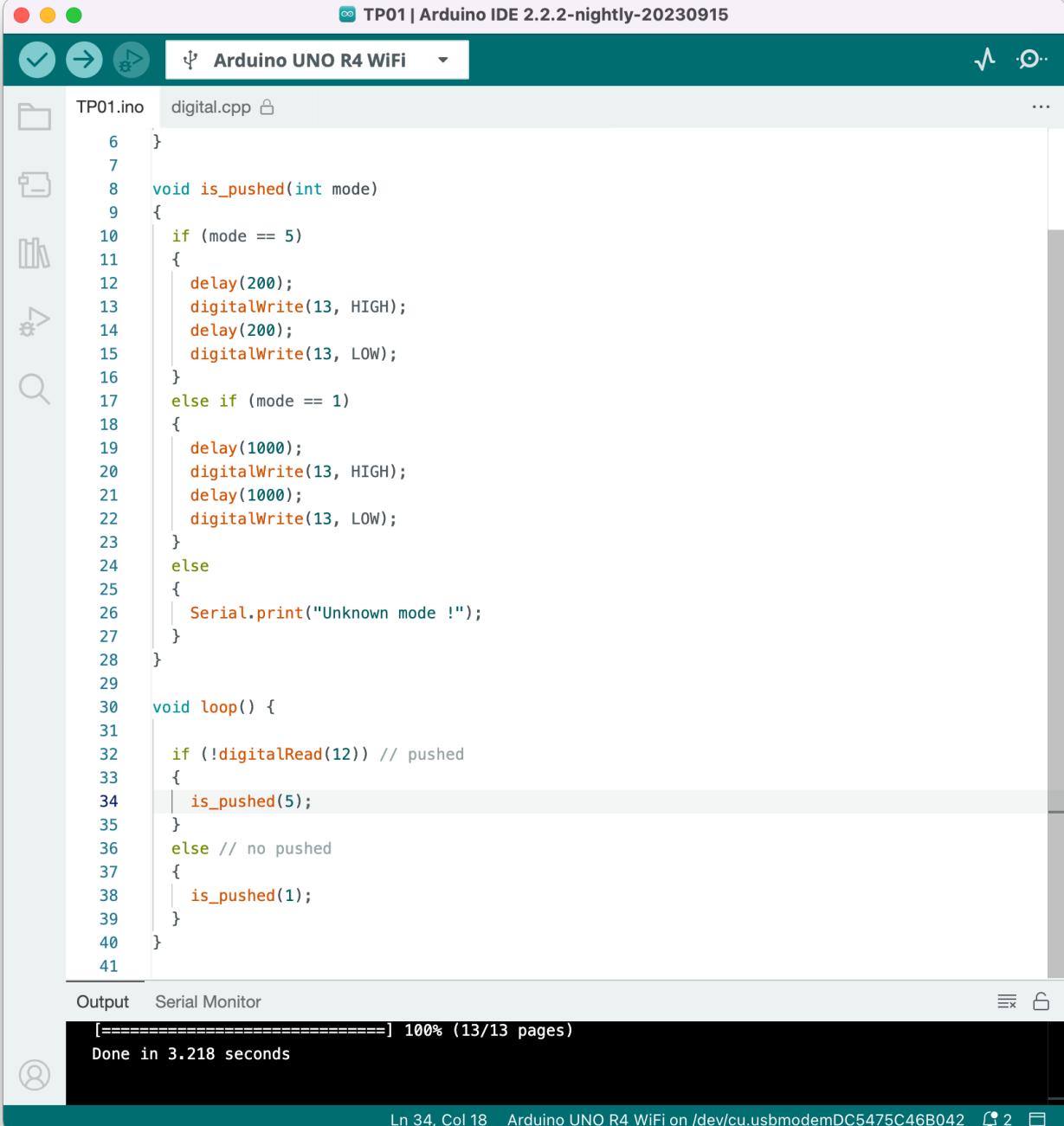
The "Output" tab is active, showing the serial monitor output:

```
[=====] 55% (7/13 pages)
[=====] 61% (8/13 pages)
[=====] 69% (9/13 pages)
[=====] 76% (10/13 pages)
2 New Notifications
```

The status bar at the bottom indicates "Ln 10, Col 8" and "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

Exercice 05

Le code permettant de faire varier la fréquence d'allumage de la LED en fonction de l'appui sur le bouton poussoir est le suivant :



The screenshot shows the Arduino IDE interface with the following details:

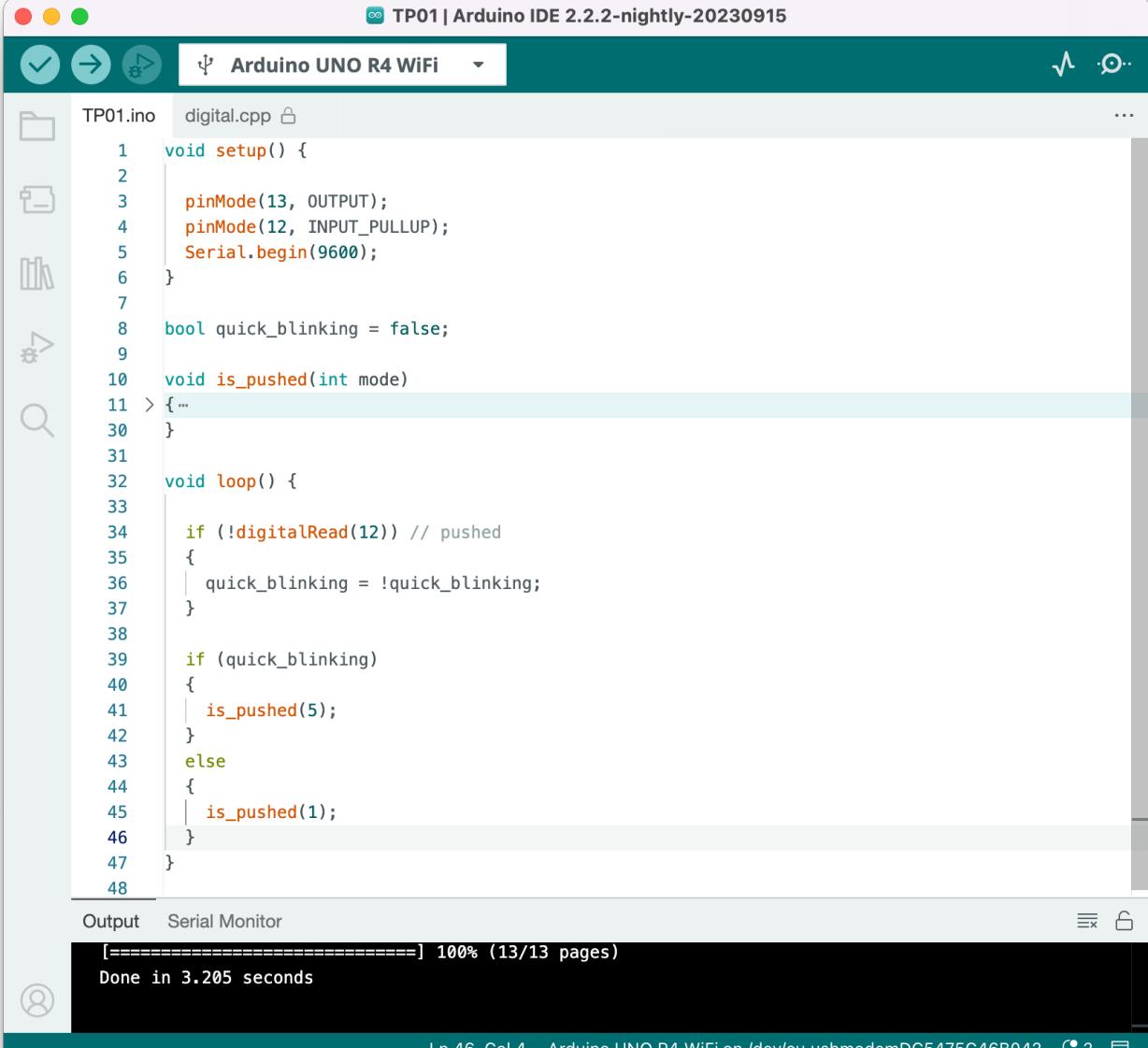
- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbox:** Includes icons for file operations (New, Open, Save, Print), build (Build, Upload, Verify), and search.
- File Explorer:** Shows the project structure with files TP01.ino and digital.cpp.
- Code Editor:** Displays the C++ code for the Arduino sketch. The code defines two functions: `is_pushed` and `loop`. The `is_pushed` function checks if a button is pushed and toggles a LED at different frequencies based on the mode (5 or 1). The `loop` function reads the button state and calls `is_pushed` accordingly.

```
TP01.ino digital.cpp
6 }
7
8 void is_pushed(int mode)
9 {
10 if (mode == 5)
11 {
12     delay(200);
13     digitalWrite(13, HIGH);
14     delay(200);
15     digitalWrite(13, LOW);
16 }
17 else if (mode == 1)
18 {
19     delay(1000);
20     digitalWrite(13, HIGH);
21     delay(1000);
22     digitalWrite(13, LOW);
23 }
24 else
25 {
26     Serial.print("Unknown mode !");
27 }
28 }
29
30 void loop() {
31
32     if (!digitalRead(12)) // pushed
33     {
34         is_pushed(5);
35     }
36     else // no pushed
37     {
38         is_pushed(1);
39     }
40 }
```

- Output Panel:** Shows the compilation progress: [=====] 100% (13/13 pages) and Done in 3.218 seconds.
- Status Bar:** Shows the current line (Ln 34, Col 18), board (Arduino UNO R4 WiFi), port (/dev/cu.usbmodemDC5475C46B042), and other system information.

Exercice 06

Le code permettant de faire changer la fréquence de clignotement après un appui bref est le suivant.



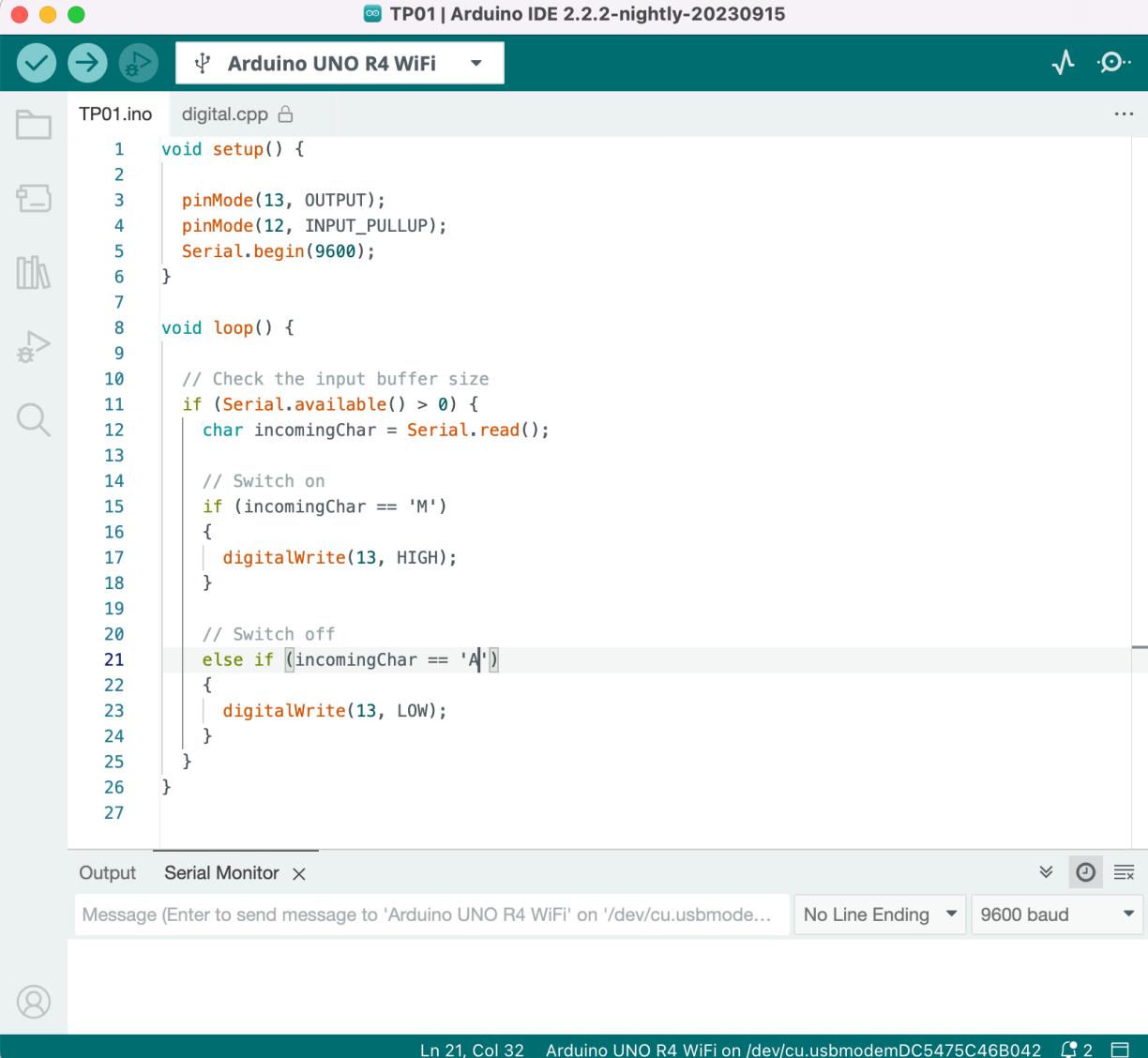
The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The central editor window displays the code for "TP01.ino" under the file "digital.cpp". The code implements a logic to change the blink frequency based on a button press. The "Output" tab at the bottom shows the compilation progress: "[=====] 100% (13/13 pages)" and "Done in 3.205 seconds". The status bar at the bottom right indicates "Ln 46, Col 4" and "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

```
1 void setup() {
2
3     pinMode(13, OUTPUT);
4     pinMode(12, INPUT_PULLUP);
5     Serial.begin(9600);
6 }
7
8 bool quick_blinking = false;
9
10 void is_pushed(int mode)
11 > { ...
12 }
13
14 void loop() {
15
16     if (!digitalRead(12)) // pushed
17     {
18         quick_blinking = !quick_blinking;
19     }
20
21     if (quick_blinking)
22     {
23         is_pushed(5);
24     }
25     else
26     {
27         is_pushed(1);
28     }
29 }
30
31 }
```

Le défaut de notre programme est le fait que, surtout pendant le mode 1 Hz est activé, notre appui sur le bouton poussoir n'est pas détecté : nous appuyons dessus lors d'un `delay(1000)`, et ne capturons donc pas le signal envoyé par le bouton en passant par la fonction `digitalRead()`. Il faut donc appuyer au bon moment, ou rester appuyé, ce qui n'est pas précis.

Exercice 07

Le code permettant d'écouter les deux lettres du clavier A et M est le suivant :



The screenshot shows the Arduino IDE interface with the following details:

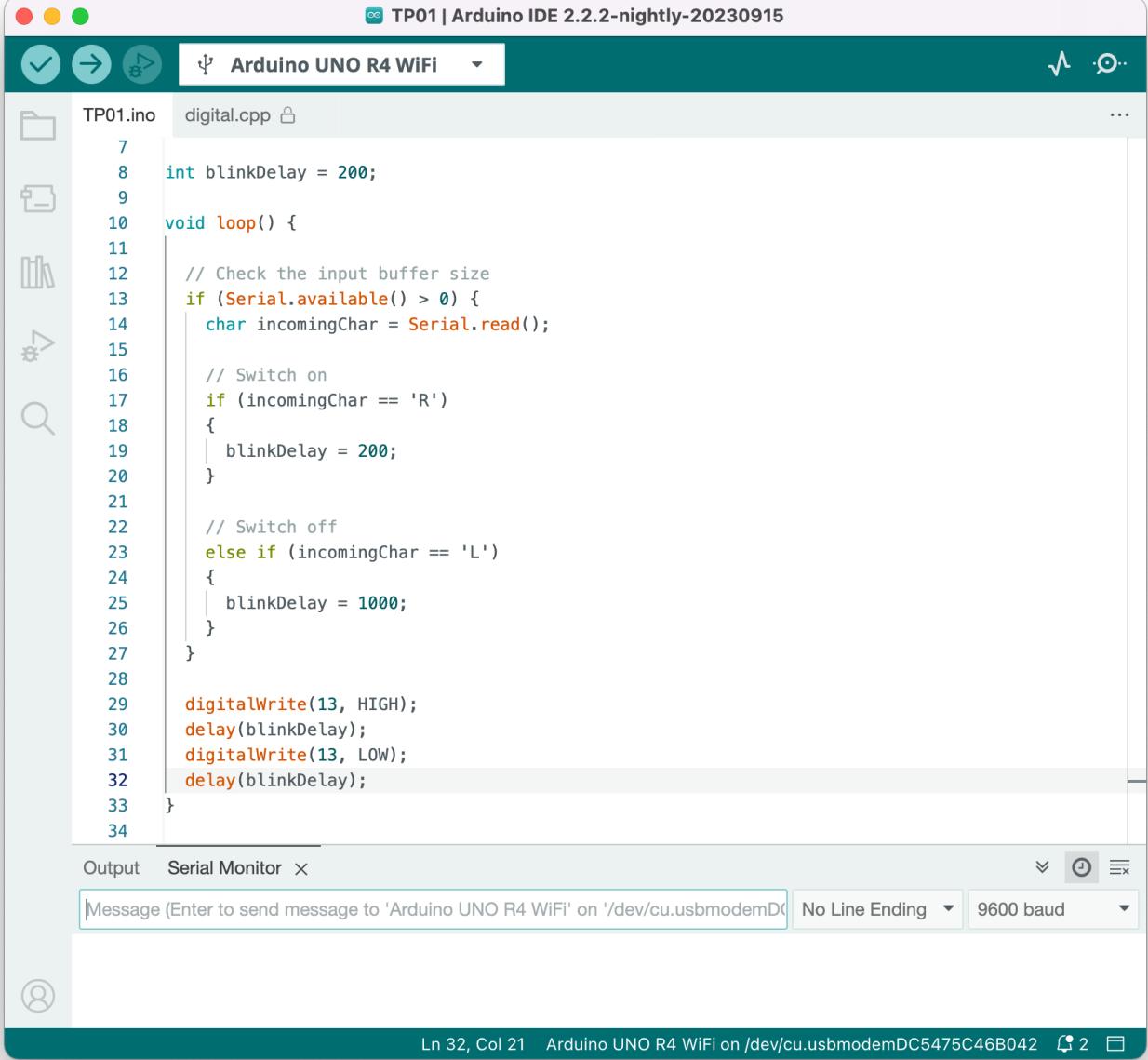
- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbars:** Standard Arduino IDE toolbars for file operations.
- File Explorer:** Shows the project structure with files TP01.ino and digital.cpp.
- Code Editor:** Displays the following C++ code for an Arduino sketch:

```
TP01.ino digital.cpp
1 void setup() {
2
3     pinMode(13, OUTPUT);
4     pinMode(12, INPUT_PULLUP);
5     Serial.begin(9600);
6 }
7
8 void loop() {
9
10    // Check the input buffer size
11    if (Serial.available() > 0) {
12        char incomingChar = Serial.read();
13
14        // Switch on
15        if (incomingChar == 'M')
16        {
17            digitalWrite(13, HIGH);
18        }
19
20        // Switch off
21        else if (incomingChar == 'A')
22        {
23            digitalWrite(13, LOW);
24        }
25    }
26 }
27
```

- Output/Serial Monitor:** Shows the serial port configuration: "Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodemDC5475C46B042'" and baud rate "9600 baud".
- Status Bar:** Shows the current line and column: "Ln 21, Col 32" and the connected port: "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

Exercice 08

Le code permettant de contrôler le clignotement par l'ordinateur est le suivant :



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Board Selection:** Arduino UNO R4 WiFi
- Code Editor:** The file `TP01.ino` contains the following C++ code:

```
int blinkDelay = 200;
void loop() {
    // Check the input buffer size
    if (Serial.available() > 0) {
        char incomingChar = Serial.read();

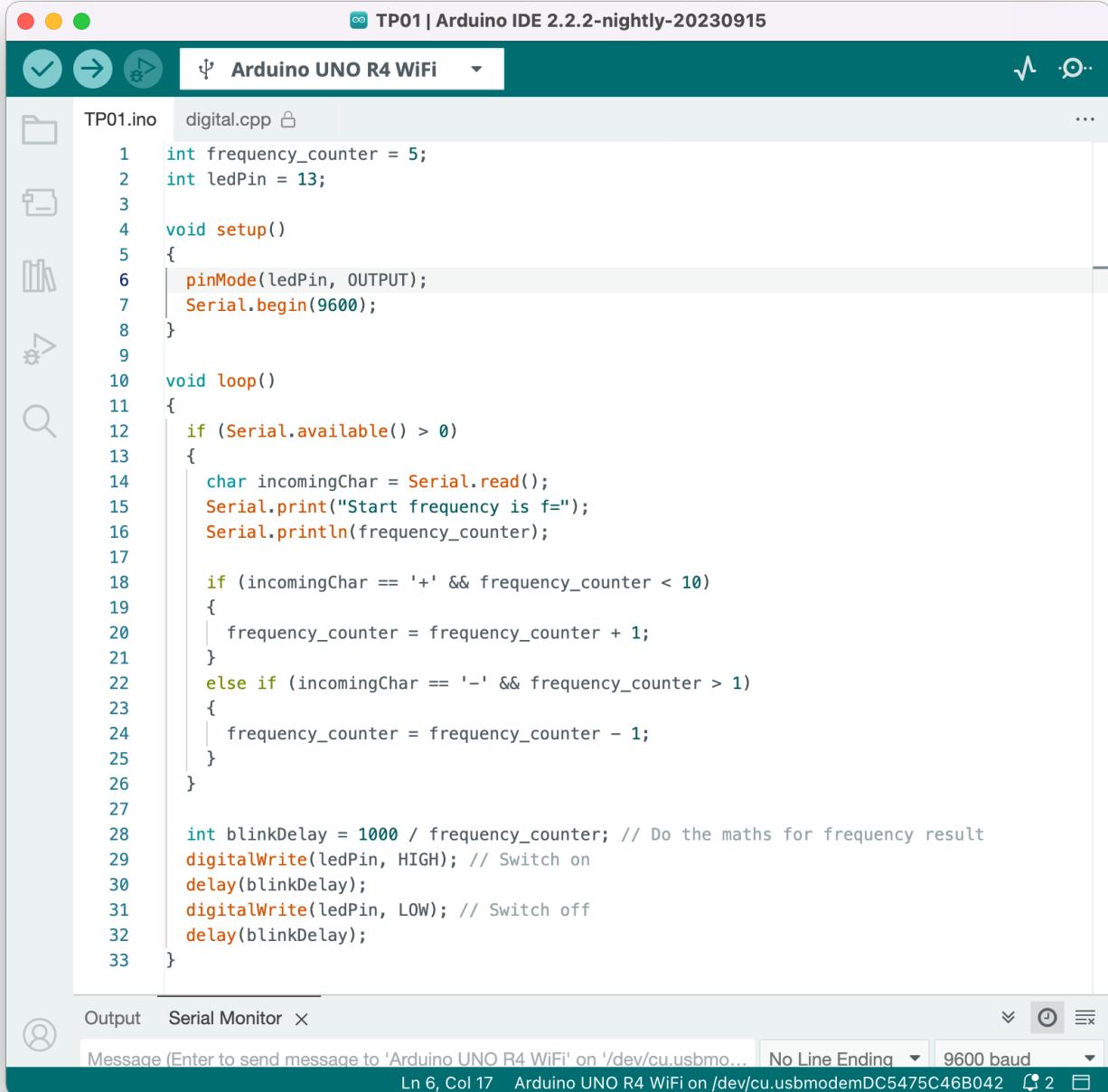
        // Switch on
        if (incomingChar == 'R') {
            blinkDelay = 200;
        }

        // Switch off
        else if (incomingChar == 'L') {
            blinkDelay = 1000;
        }
    }

    digitalWrite(13, HIGH);
    delay(blinkDelay);
    digitalWrite(13, LOW);
    delay(blinkDelay);
}
```
- Output Tab:** Shows the message "Message (Enter to send message to 'Arduino UNO R4 WiFi' on '/dev/cu.usbmodemDC5475C46B042' No Line Ending 9600 baud".
- Status Bar:** Ln 32, Col 21 Arduino UNO R4 WiFi on /dev/cu.usbmodemDC5475C46B042 2

Exercice 09

Le code permettant de modifier la fréquence de la LED à l'aide des touches + et – est le suivant :



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbox:** Includes icons for file operations (New, Open, Save, Print), undo/redo, and other common functions.
- Code Editor:** Displays the `TP01.ino` file content. The code is as follows:

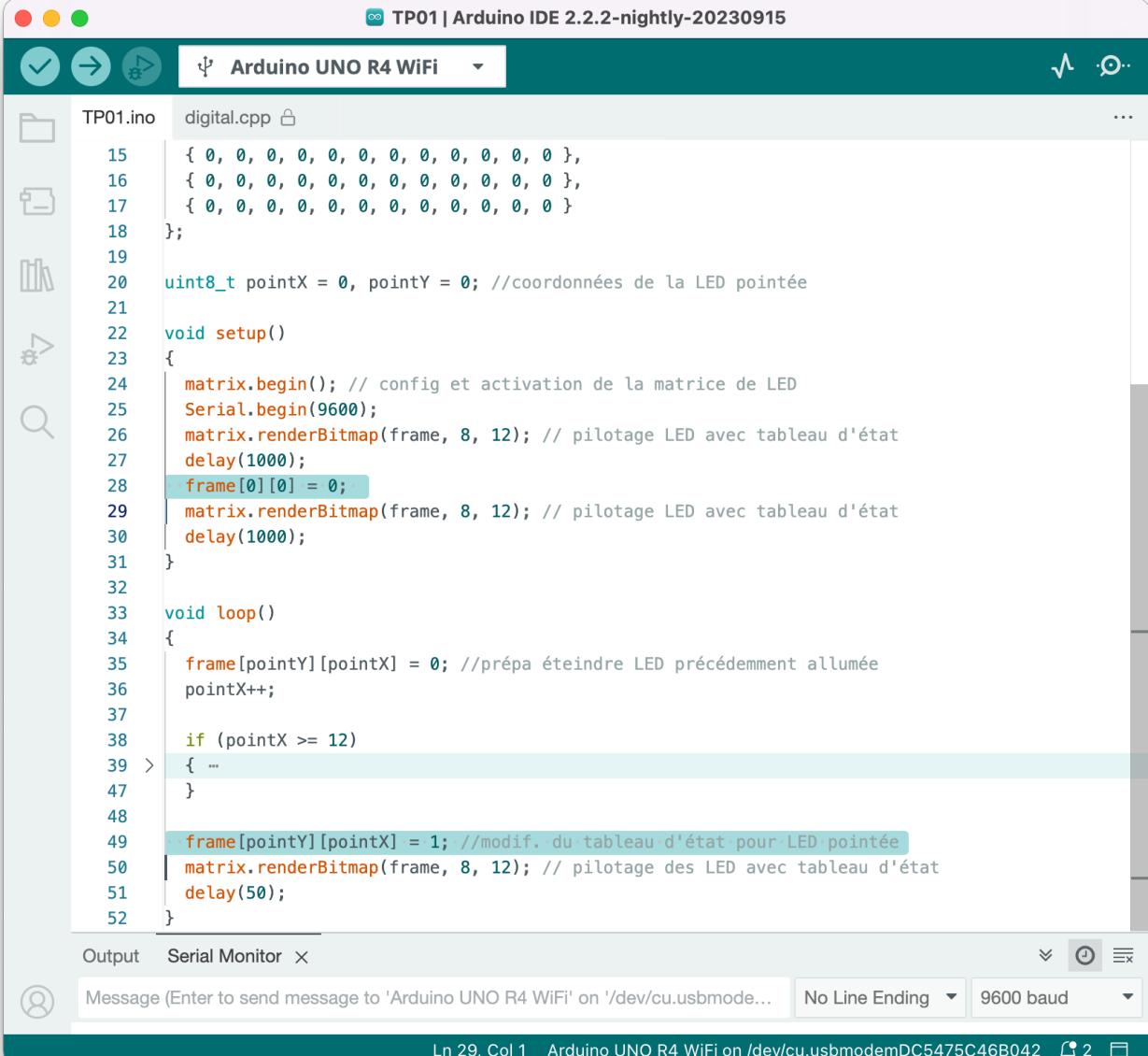
```
TP01.ino digital.cpp
1 int frequency_counter = 5;
2 int ledPin = 13;
3
4 void setup()
{
5     pinMode(ledPin, OUTPUT);
6     Serial.begin(9600);
7 }
8
9
10 void loop()
{
11     if (Serial.available() > 0)
12     {
13         char incomingChar = Serial.read();
14         Serial.print("Start frequency is f=");
15         Serial.println(frequency_counter);
16
17         if (incomingChar == '+' && frequency_counter < 10)
18         {
19             frequency_counter = frequency_counter + 1;
20         }
21         else if (incomingChar == '-' && frequency_counter > 1)
22         {
23             frequency_counter = frequency_counter - 1;
24         }
25     }
26
27     int blinkDelay = 1000 / frequency_counter; // Do the maths for frequency result
28     digitalWrite(ledPin, HIGH); // Switch on
29     delay(blinkDelay);
30     digitalWrite(ledPin, LOW); // Switch off
31     delay(blinkDelay);
32 }
33 }
```

- Output Tab:** Shows the serial port configuration: "Message (Enter to send message to 'Arduino UNO R4 WiFi' on '/dev/cu.usbmo...' No Line Ending 9600 baud". It also displays the current line number (Ln 6, Col 17) and the connection status (Arduino UNO R4 WiFi on /dev/cu.usbmodemDC5475C46B042).

Exercice 11

Question 1

Le programme permettant de n'afficher qu'une seule LED à la fois est le suivant. Il suffit de ne modifier qu'une seule valeur dans le tableau d'état. (modifications des lignes 28 et 49).



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- Toolbox:** Includes icons for file operations (New, Open, Save, Print), build (Build, Build All, Upload), and preferences.
- File Explorer:** Shows a folder named "TP01.ino" containing "digital.cpp".
- Code Editor:** Displays the following C++ code for an 8x12 LED matrix:

```
TP01.ino  digital.cpp

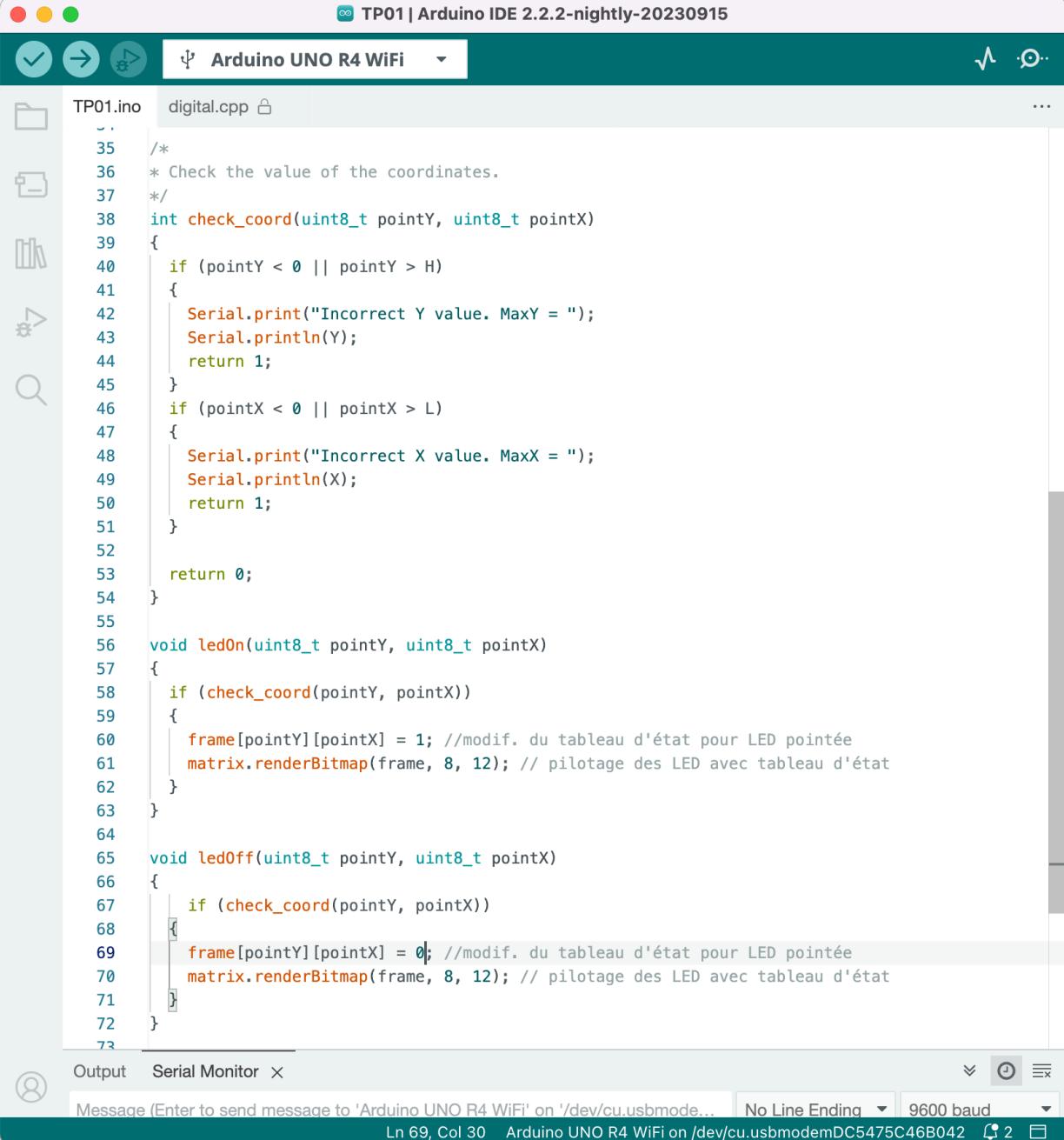
15 { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 },
16 { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 },
17 { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 }
18 };

19
20 uint8_t pointX = 0, pointY = 0; // coordonnées de la LED pointée
21
22 void setup()
23 {
24     matrix.begin(); // config et activation de la matrice de LED
25     Serial.begin(9600);
26     matrix.renderBitmap(frame, 8, 12); // pilotage LED avec tableau d'état
27     delay(1000);
28     frame[0][0] = 0;
29     matrix.renderBitmap(frame, 8, 12); // pilotage LED avec tableau d'état
30     delay(1000);
31 }
32
33 void loop()
34 {
35     frame[pointY][pointX] = 0; // prépa éteindre LED précédemment allumée
36     pointX++;
37
38     if (pointX >= 12)
39     {
40         ...
41     }
42
43     frame[pointY][pointX] = 1; // modif. du tableau d'état pour LED pointée
44     matrix.renderBitmap(frame, 8, 12); // pilotage des LED avec tableau d'état
45     delay(50);
46 }
```
- Output Tab:** Shows "Serial Monitor" tab.
- Serial Monitor:** A text input field with placeholder "Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodem...'") and settings for "No Line Ending" and "9600 baud".
- Status Bar:** Shows "Ln 29, Col 1" and "Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042".

Question 2

Voici le code des deux fonctions `ledOn()` et `ledOff()` permettant d'allumer et d'éteindre une LED à partir de coordonnées `x`, `y` données.

Note : La fonction `check_coord()` a pour but de vérifier que les coordonnées entrées dans la fonction ne sont pas en dehors du tableau d'état.



The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The central area displays the code for "TP01.ino" which includes "digital.cpp". The code defines two functions: `check_coord` for coordinate validation and `ledOn` and `ledOff` for LED control. It also includes a `frame` array and a `matrix` object for bitmap rendering.

```
TP01.ino digital.cpp

35 /*
36 * Check the value of the coordinates.
37 */
38 int check_coord(uint8_t pointY, uint8_t pointX)
39 {
40     if (pointY < 0 || pointY > H)
41     {
42         Serial.print("Incorrect Y value. MaxY = ");
43         Serial.println(Y);
44         return 1;
45     }
46     if (pointX < 0 || pointX > L)
47     {
48         Serial.print("Incorrect X value. MaxX = ");
49         Serial.println(X);
50         return 1;
51     }
52
53     return 0;
54 }

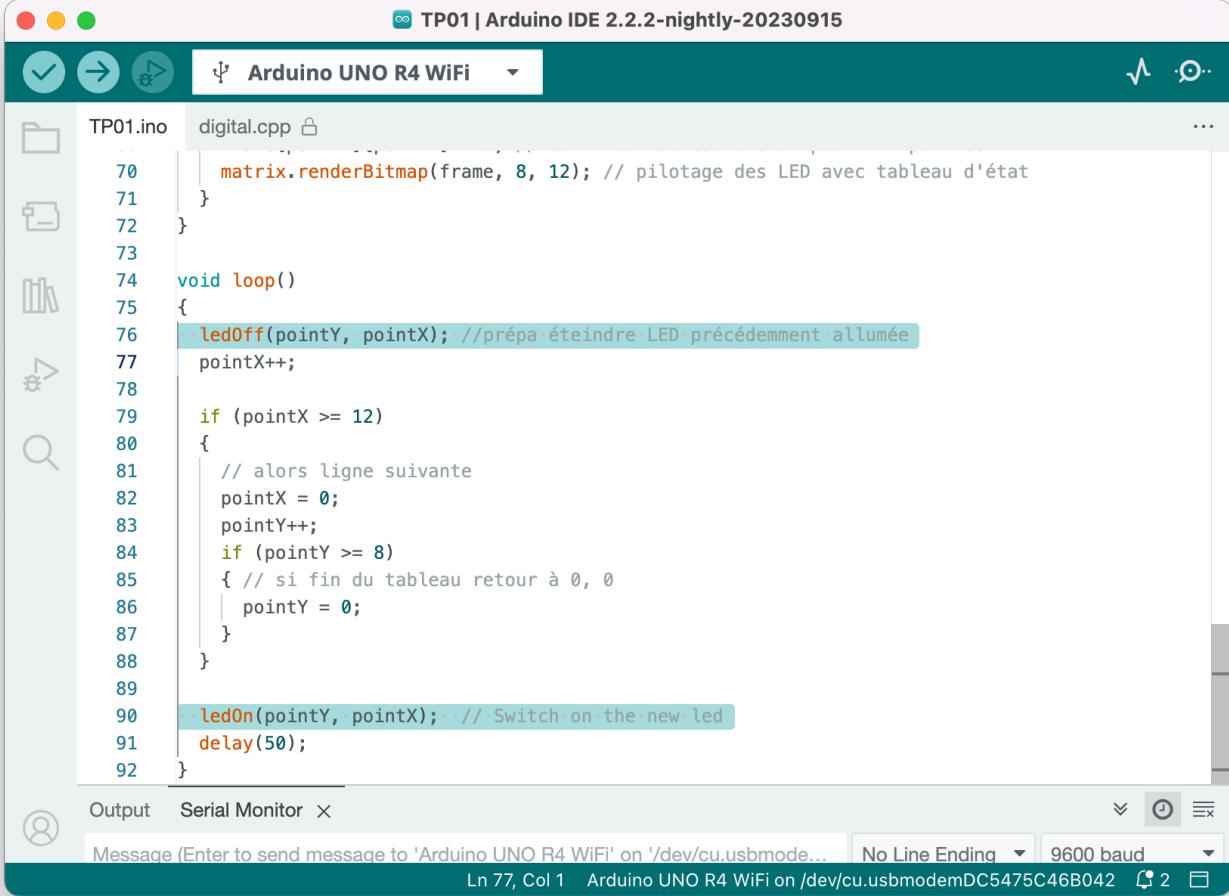
56 void ledOn(uint8_t pointY, uint8_t pointX)
57 {
58     if (check_coord(pointY, pointX))
59     {
60         frame[pointY][pointX] = 1; //modif. du tableau d'état pour LED pointée
61         matrix.renderBitmap(frame, 8, 12); // pilotage des LED avec tableau d'état
62     }
63 }

65 void ledOff(uint8_t pointY, uint8_t pointX)
66 {
67     if (check_coord(pointY, pointX))
68     {
69         frame[pointY][pointX] = 0; //modif. du tableau d'état pour LED pointée
70         matrix.renderBitmap(frame, 8, 12); // pilotage des LED avec tableau d'état
71     }
72 }

73 
```

The bottom status bar shows "Message (Enter to send message to 'Arduino Uno R4 WiFi' on '/dev/cu.usbmodem...' No Line Ending 9600 baud Ln 69, Col 30 Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042 2" and a terminal window icon.

L'utilisation de ces deux fonctions est la suivante :



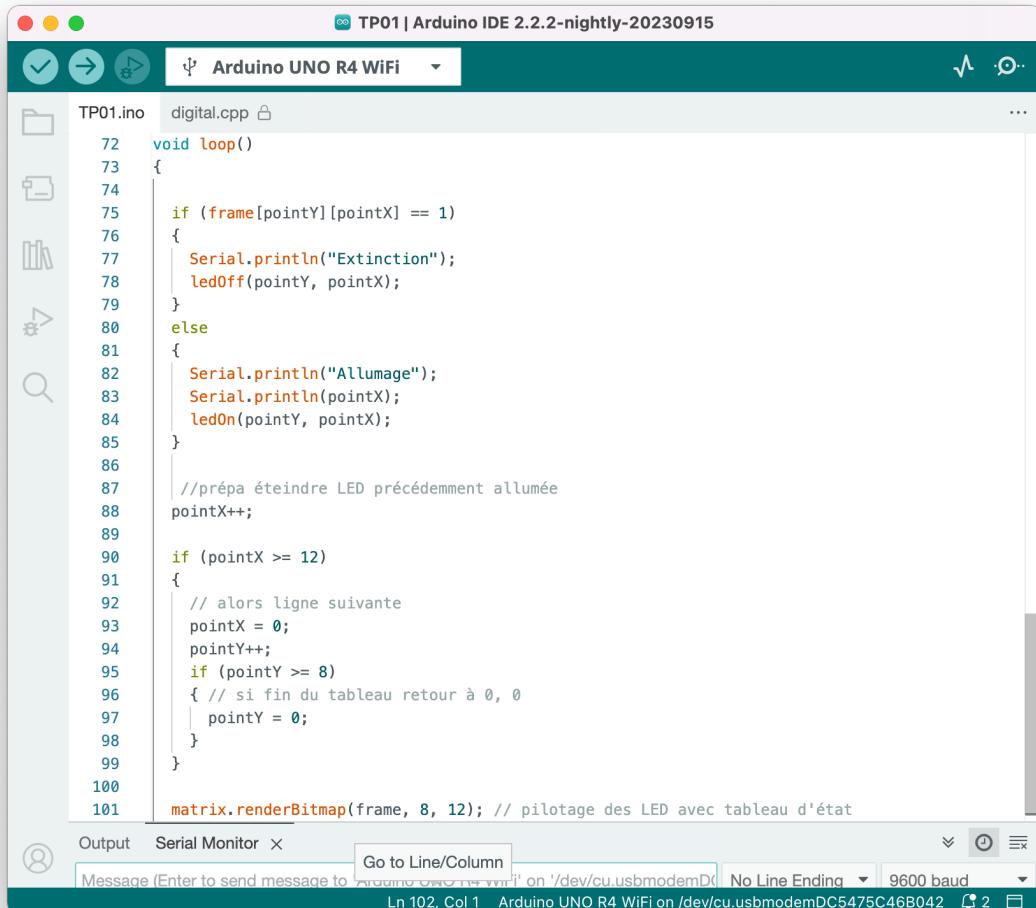
The screenshot shows the Arduino IDE interface with the title bar "TP01 | Arduino IDE 2.2.2-nightly-20230915". The central workspace displays the following C++ code for an Arduino Uno R4 WiFi:

```
TP01.ino digital.cpp
70     matrix.renderBitmap(frame, 8, 12); // pilotage des LED avec tableau d'état
71 }
72 }
73
74 void loop()
75 {
76     ledOff(pointY, pointX); //prépa éteindre LED précédemment allumée
77     pointX++;
78
79     if (pointX >= 12)
80     {
81         // alors ligne suivante
82         pointX = 0;
83         pointY++;
84         if (pointY >= 8)
85         { // si fin du tableau retour à 0, 0
86             pointY = 0;
87         }
88     }
89
90     ledOn(pointY, pointX); // Switch on the new led
91     delay(50);
92 }
```

The code implements a dot-matrix display using the `matrix` library. It defines a `loop()` function that iterates through a 12x8 grid of LEDs. For each LED at position `(pointY, pointX)`, it first turns it off (`ledOff`) and then turns it on again (`ledOn`). After processing all LEDs in a row, it moves to the next row. If it reaches the bottom of the grid, it loops back to the top of the first row.

Question 3

Dans ce code, l'écran de LED est progressivement rempli en allumant chaque LED, puis il est progressivement vidé en éteignant chaque LED.



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** TP01 | Arduino IDE 2.2.2-nightly-20230915
- File Explorer:** Shows TP01.ino and digital.cpp.
- Code Editor:** Displays the following C++ code for an Arduino Uno R4 WiFi:

```
TP01.ino digital.cpp
72 void loop()
73 {
74     if (frame[pointY][pointX] == 1)
75     {
76         Serial.println("Extinction");
77         ledOff(pointY, pointX);
78     }
79     else
80     {
81         Serial.println("Allumage");
82         Serial.println(pointX);
83         ledOn(pointY, pointX);
84     }
85
86     //prépa éteindre LED précédemment allumée
87     pointX++;
88
89     if (pointX >= 12)
90     {
91         // alors ligne suivante
92         pointX = 0;
93         pointY++;
94         if (pointY >= 8)
95         { // si fin du tableau retour à 0, 0
96             pointY = 0;
97         }
98     }
99 }
100
101 matrix.renderBitmap(frame, 8, 12); // pilotage des LED avec tableau d'état
```
- Output Tab:** Shows the serial monitor output:

```
Message (Enter to send message to Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042) on '/dev/cu.usbmodemDC5475C46B042' No Line Ending 9600 baud
Ln 102, Col 1 Arduino Uno R4 WiFi on /dev/cu.usbmodemDC5475C46B042
```

