

La boucle « for »

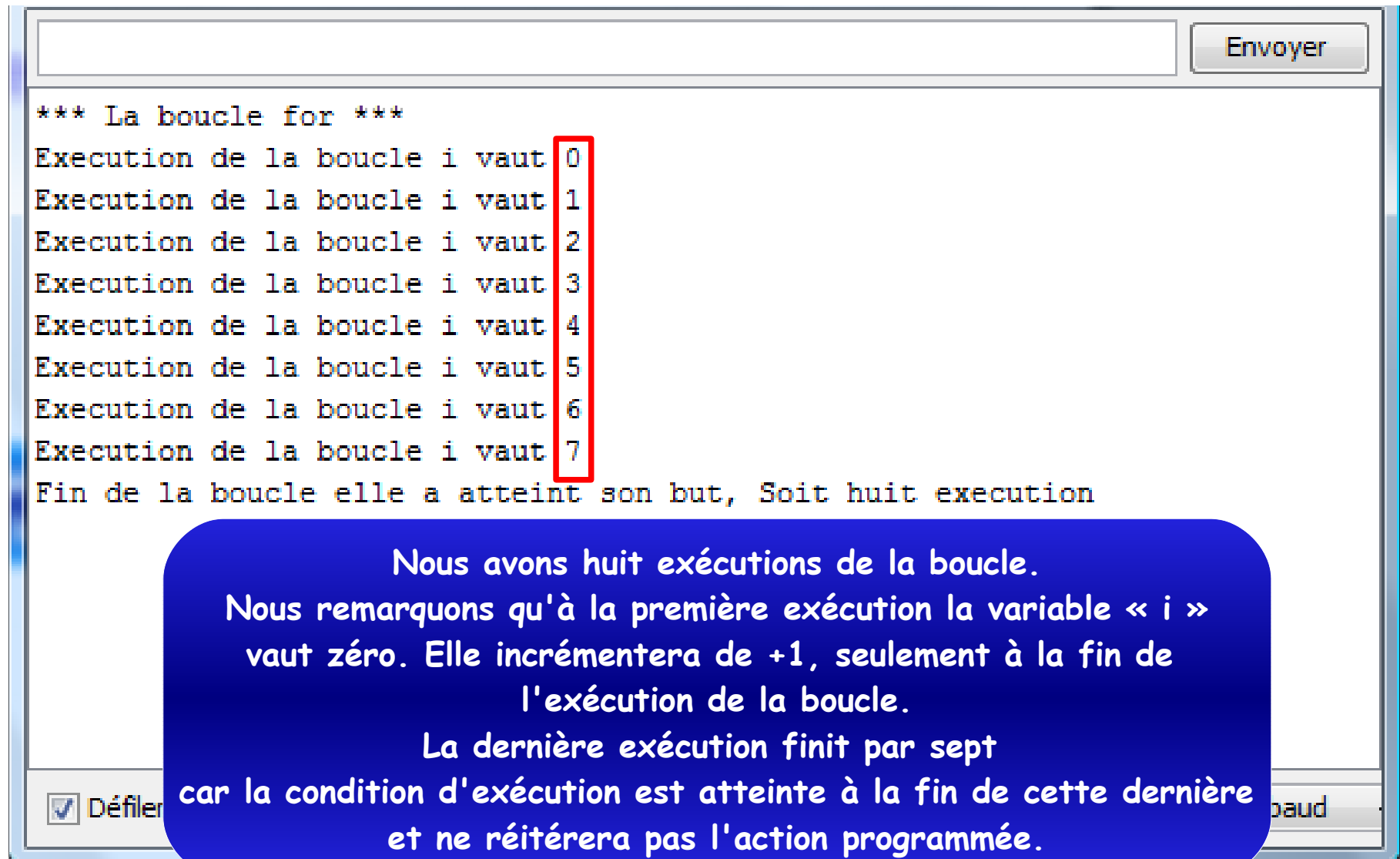
Voici notre programme
Avec une boucle « **for** »

```
Fichier Édition Croquis Outils Aide
1 /*LAISSEZ ARDUINO*/
2 /*La Boucle for*/
3 /*Travaux de mise en pratique*/
4
5 void setup() {
6   Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7   Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8 }
9
10 void loop() {
11   for(int i=0; i<8; i++){//Mise en place de la boucle.
12     Serial.print("Execution de la boucle i vaut ");//Affichage d'un message complémentaire a la valeur de "i"
13     Serial.println(i);//Affichage de la valeur de "i".
14     |
15     delay(2000);//Arret temporisé du programme pour une lecture des executions.
16   }
17   Serial.print("Fin de la boucle elle a atteint son but, Soit huit execution");
18   while(1);// Arret forcé du programme.
19 }//fin de loop.
20
```

Téléversement terminé

Le croquis utilise 2 506 octets (7%) de l'espace de stockage de programmes. Le maximum est de 32 256 octets.
Les variables globales utilisent 314 octets (15%) de mémoire dynamique, ce qui laisse 1 734 octets pour les variables

La boucle « for »



```
*** La boucle for ***
Execution de la boucle i vaut 0
Execution de la boucle i vaut 1
Execution de la boucle i vaut 2
Execution de la boucle i vaut 3
Execution de la boucle i vaut 4
Execution de la boucle i vaut 5
Execution de la boucle i vaut 6
Execution de la boucle i vaut 7
Fin de la boucle elle a atteint son but, Soit huit execution
```

Nous avons huit exécutions de la boucle.
Nous remarquons qu'à la première exécution la variable « i »
vaut zéro. Elle incrémentera de +1, seulement à la fin de
l'exécution de la boucle.
La dernière exécution finit par sept
car la condition d'exécution est atteinte à la fin de cette dernière
et ne réitérera pas l'action programmée.

La boucle « for »

Modifions le retour d'exécution par « i+2 »
et voyons le résultat.

```
Fichier Édition Croquis Outils Aide
[Icons]
for$
1 /*PLAISIR ARDUINO*/
2 /*La Boucle for*/
3 /*Travaux de mise en pratique*/
4
5 void setup() {
6   Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7   Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8 }
9
10 void loop() {
11   for(int i= 0; i< 8; i= i+2) { //Mise en place de la boucle.
12     Serial.print("Execution de la boucle i vaut "); //Affichage d'un message complémentaire a la valeur de "i"
13     Serial.println(i); //Affichage de la valeur de "i".
14
15     //delay(2000); //Arret temporisé du programme pour une lecture et compréhension des exécutions.
16   }
17   Serial.print("Fin de la boucle elle a atteint son but.");
18   while(1); // Arrêt forcé du programme.
19 } //fin de loop.
20
```

1

Arduino/Genuino Uno on COM5

La boucle « for »

Envoyer

*** La boucle for ***

Execution de la boucle i vaut 0.00

Execution de la boucle i vaut 2.00

Execution de la boucle i vaut 4.00

Execution de la boucle i vaut 6.00

Fin de la boucle elle a atteint son but.

Nous n'avons plus que quatre exécutions.
La valeur de « i » incrémente de +2
à la fin de chaque exécution.

☒ Défilement automatique

Pas de fin de ligne ▼

9600 baud

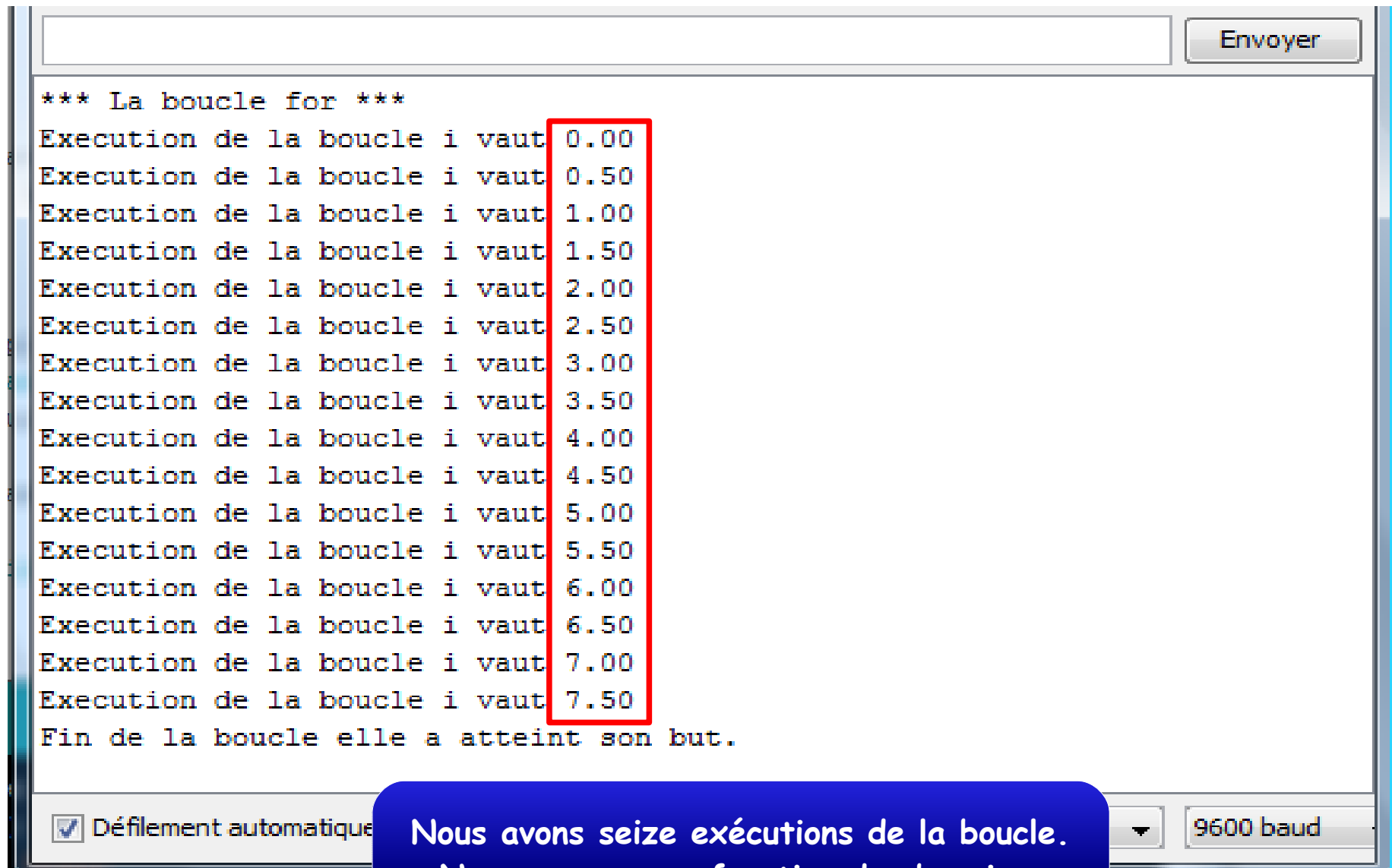
La boucle « for »

Modifions le type et le retour d'exécution de la variable de contrôle et voyons le résultat.

for \$

```
1  /*PLAISIR ARDUINO*/
2  /*La Boucle for*/
3  /*Travaux de mise en
4
5  void setup() {
6    Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7    Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8  }
9
10 void loop() {
11   for(float i= 0; i< 8; i= i+0.5){//Mise en place de la boucle.
12     Serial.print("Execution de la boucle i vaut ");//Affichage d'un message complementaire a la valeur de "i"
13     Serial.println(i);//Affichage de la valeur de "i".
14
15     //delay(2000);//Arret temporisé du programme pour une lecture et comprehension des executions.
16   }
17   Serial.print("Fin de la boucle elle a atteint son but.");
18   while(1);// Arret forcé du programme.
19 }//fin de loop.
20
```

La boucle « for »



```
*** La boucle for ***
Execution de la boucle i vaut 0.00
Execution de la boucle i vaut 0.50
Execution de la boucle i vaut 1.00
Execution de la boucle i vaut 1.50
Execution de la boucle i vaut 2.00
Execution de la boucle i vaut 2.50
Execution de la boucle i vaut 3.00
Execution de la boucle i vaut 3.50
Execution de la boucle i vaut 4.00
Execution de la boucle i vaut 4.50
Execution de la boucle i vaut 5.00
Execution de la boucle i vaut 5.50
Execution de la boucle i vaut 6.00
Execution de la boucle i vaut 6.50
Execution de la boucle i vaut 7.00
Execution de la boucle i vaut 7.50
Fin de la boucle elle a atteint son but.
```

☒ Défilement automatique

9600 baud

Nous avons seize exécutions de la boucle.
Nous pouvons en fonction des besoins
modifier le type de la variable de travail.

La boucle « for »

Modifions à présent l'initialisation de la variable.
Et voyons le résultat.

```
Fichier Édition Croquis Outils Aide
for $
1 /*PLAIS
2 /*La Bo
3 /*Trava
4
5 void setu
6 Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7 Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8 }
9
10 void loop() {
11   for(float i= 5; i< 8; i= i+0.5){//Mise en place de la boucle.
12     Serial.print("Execution de la boucle i vaut ");//Affichage d'un message complementaire a la valeur de "i"
13     Serial.println(i);//Affichage de la valeur de "i".
14
15     //delay(2000);//Arret temporisé du programme pour une lecture et comprehension des executions.
16   }
17   Serial.print("Fin de la boucle elle a atteint son but.");
18   while(1);// Arret forcé du programme.
19 }//fin de loop.
20
```

1

Arduino/Genuino Uno on COM5

La boucle « for »

Envoyer

```
*** La boucle for ***  
Execution de la boucle i vaut 5.00  
Execution de la boucle i vaut 5.50  
Execution de la boucle i vaut 6.00  
Execution de la boucle i vaut 6.50  
Execution de la boucle i vaut 7.00  
Execution de la boucle i vaut 7.50  
Fin de la boucle elle a atteint son but.
```

La variable « i » vaut cinq à la première exécution.
Elle a été initialisé à l'entrée dans la boucle
par la valeur définie et commence son comptage depuis
cette valeur.

☒ Défilement automatique

Pas de fin de ligne ▼

9600 baud

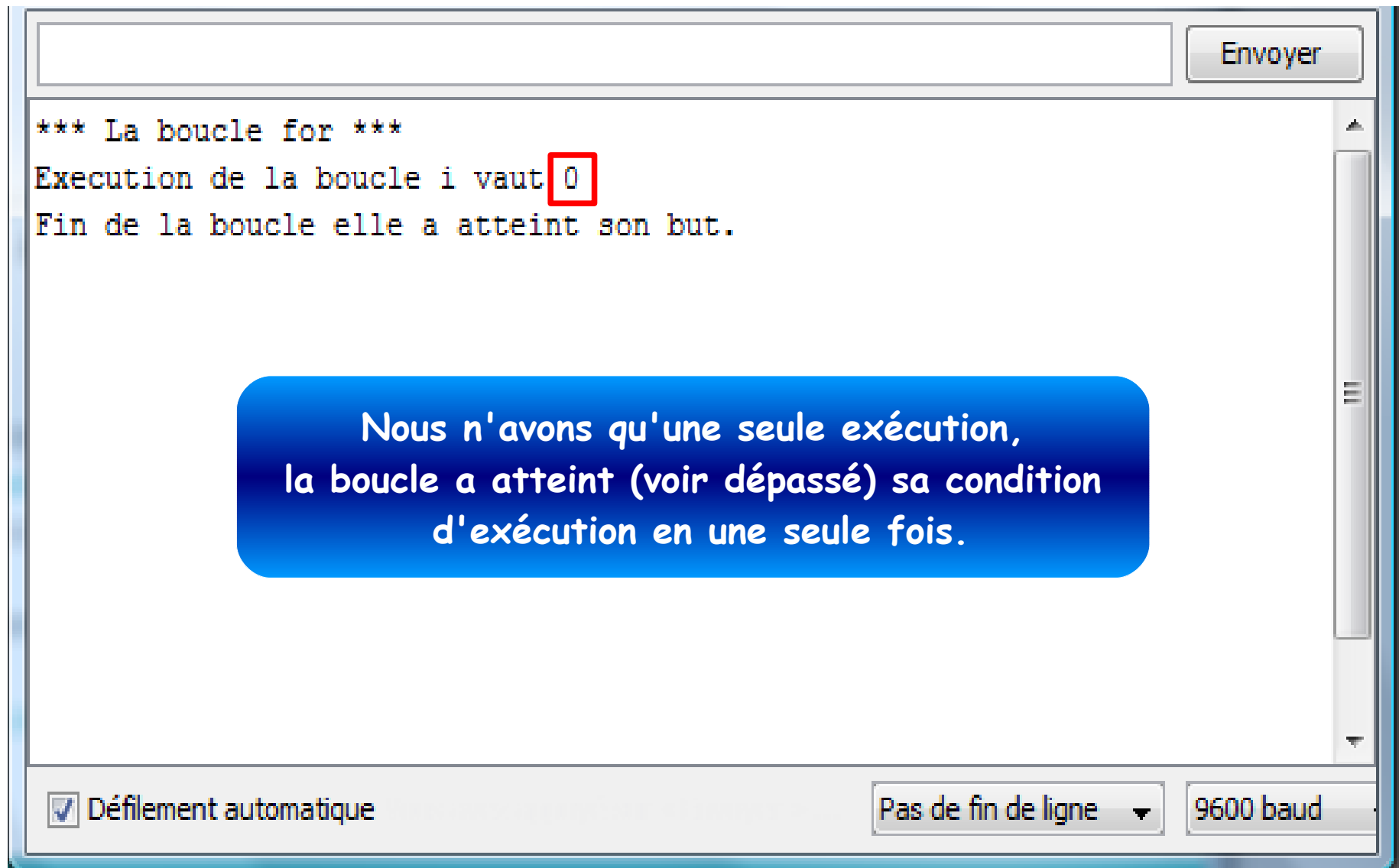
La boucle « for »

Reprenons notre première boucle et modifions maintenant notre retour d'exécution de façon exagérée.

for \$

```
1  /*PLAISIR ARDUINO*/
2  /*La Boucle for*/
3  /*Travaux de mise en pratique*/
4
5  void setup() {
6    Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7    Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8  }
9
10 void loop() {
11   for(int i= 0; i< 8; i= i+16){//Mise en place de la boucle.
12     Serial.print("Execution de la boucle i vaut ");//Affichage d'un message complementaire a la valeur de "i"
13     Serial.println(i);//Affichage de la valeur de "i".
14
15     //delay(2000);//Arret temporisé du programme pour une lecture et comprehension des executions.
16   }
17   Serial.print("Fin de la boucle elle a atteint son but.");
18   while(1);// Arret forcé du programme.
19 }//fin de loop.
20
```

La boucle « for »



La boucle « for »

A travers ces travaux pratiques,
nous constatons que ce qui compte pour la boucle s'est
d'atteindre son but, définie par la condition d'exécution.

Comment l'exploiter et pour quelle tâche ?

La boucle « for »

Voici un exemple.

```
for_EX1 $
1  /*PLAISIR ARDUINO*/
2  /*La Boucle for*/
3  /*Travaux de mise en pratique*/
4
5  void setup() {
6      Serial.begin(9600); //Ouverture et configuration de la vitesse de transmission du port serie.
7      Serial.println("*** La boucle for ***"); // Affichage au moniteur (test).
8  } // fin de setup.
9
10 void loop() {
11     //Configuration des broches de l'Arduino.
12     for (int i = 4; i < 8 ; i++) { //Mise en place de la boucle.
13         pinMode(i , OUTPUT); //Configuration des broches à la valeur de "i".
14         Serial.print("La broche "); Serial.print(i); Serial.println(" est configurée en sortie."); //Affichage du message.
15         delay(2000); //Arret temporisé du programme pour une lecture des exécutions.
16     } //fin de for.
17
18     Serial.print("Fin de la boucle elle a atteint son but");
19
20     while (1); //Arret forcé du programme.
21
22 } //fin de loop
23
```

Nous allons configurer en sortie les broches
quatre à sept .

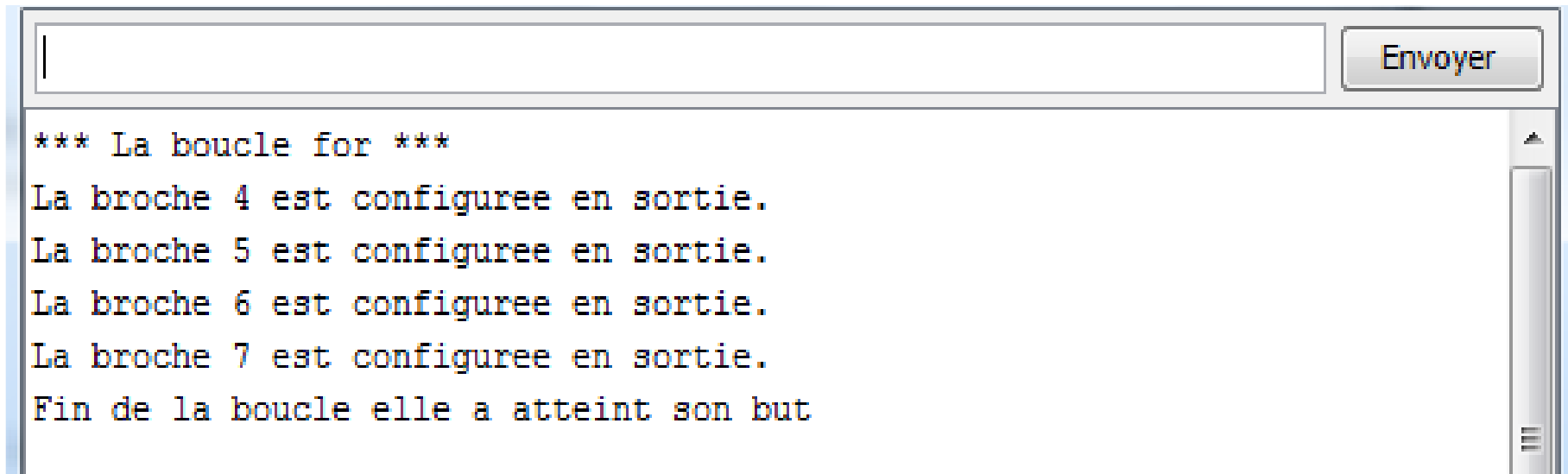
Bien-sûr cette méthode ne fonctionne que
si les broches utilisées se suivent.

Téléversement terminé

Le croquis utilise
Les variables globales

Variables locales

La boucle « for »



The screenshot shows a serial terminal window with a text input field at the top and an 'Envoyer' button. Below the input field, the output of a program is displayed in a monospaced font. The output consists of five lines: a header line '*** La boucle for ***', four lines indicating that pins 4, 5, 6, and 7 are configured as outputs, and a final line stating 'Fin de la boucle elle a atteint son but'. A scrollbar is visible on the right side of the output area.

```
*** La boucle for ***  
La broche 4 est configuree en sortie.  
La broche 5 est configuree en sortie.  
La broche 6 est configuree en sortie.  
La broche 7 est configuree en sortie.  
Fin de la boucle elle a atteint son but
```

Nous avons quatre exécutions.
La valeur de « i » est initialisée à quatre. Puis, l'action programmée configure en sortie la valeur de « i » soit la broche quatre.
Et ainsi de suite pour les broches suivantes.
Vous pouvez également l'utiliser pour les entrées.

!! ATTENTION !!

Ne pas configurer les broches zéro et une car elles sont utiles aux transmissions de données séries via « USB ».
Donc si vous utilisez cette méthode, initialisez au minimum à deux.

La boucle « for »

Il y d'autres façons pratiques d'exploiter la boucle « for »
Vous les découvrirez au fur et à mesure de votre progression.

!! BRAVO !!

A travers ces travaux pratiques, vous connaissez
son principe de fonctionnement.

FIN