

Transition Scratch/Python

Séminaire « Mathématiques et numérique »

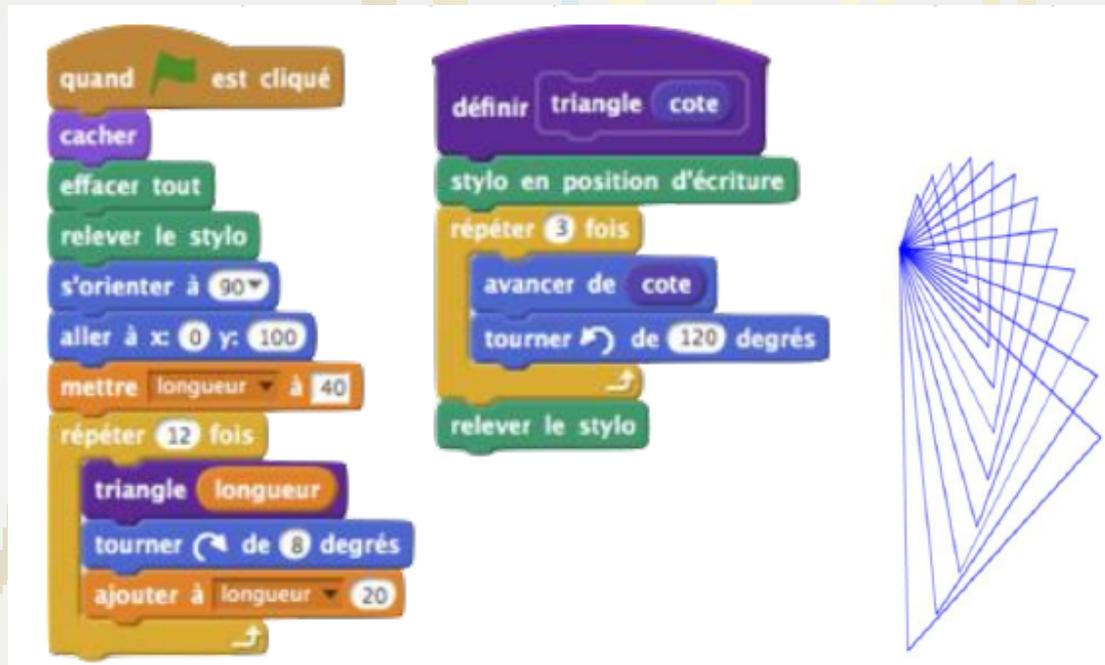
29 janvier 2020

Jérôme Michaud-Bonnet

Transition Scratch/Python

Ressources pour le lycée, juin 2017

« Les deux langages comportent, au-delà des différences évidentes de forme, des similitudes qui facilitent la transition. »



```
import turtle

def figure():

    turtle.hideturtle()
    turtle.clear()
    turtle.up()
    turtle.setheading(0)
    turtle.goto(0,100)
    longueur = 40
    for i in range(12):
        triangle(longueur)
        turtle.right(8)
        longueur = longueur + 20

def triangle(cote):
    turtle.down()
    for i in range(3):
        turtle.forward(cote)
        turtle.left(120)
    turtle.up()
```

Transition Scratch/Python

Comparaison Scratch et Python

- Un objectif commun (même **théorie**) : décomposer un problème
- Deux **technologies** différentes (paradigme objets/fonctions)
- Des **techniques** différentes (spécificités des langages)
- Des **tâches** qui peuvent être identiques

Transition Scratch/Python

Comparaison Scratch et Python

Deux **technologies** différentes :



Décomposer un problème en plusieurs objets. Décomposer les actions de chaque objet en événements parallèles



Décomposer un problème avec des fonctions

Transition Scratch/Python

Comparaison Scratch et Python

Deux **technologies** différentes :



Décomposer un problème en plusieurs objets. Décomposer les actions de chaque objet en événements parallèles

Blocs Scratch (pas de valeur renvoyée) : action, le « faire »

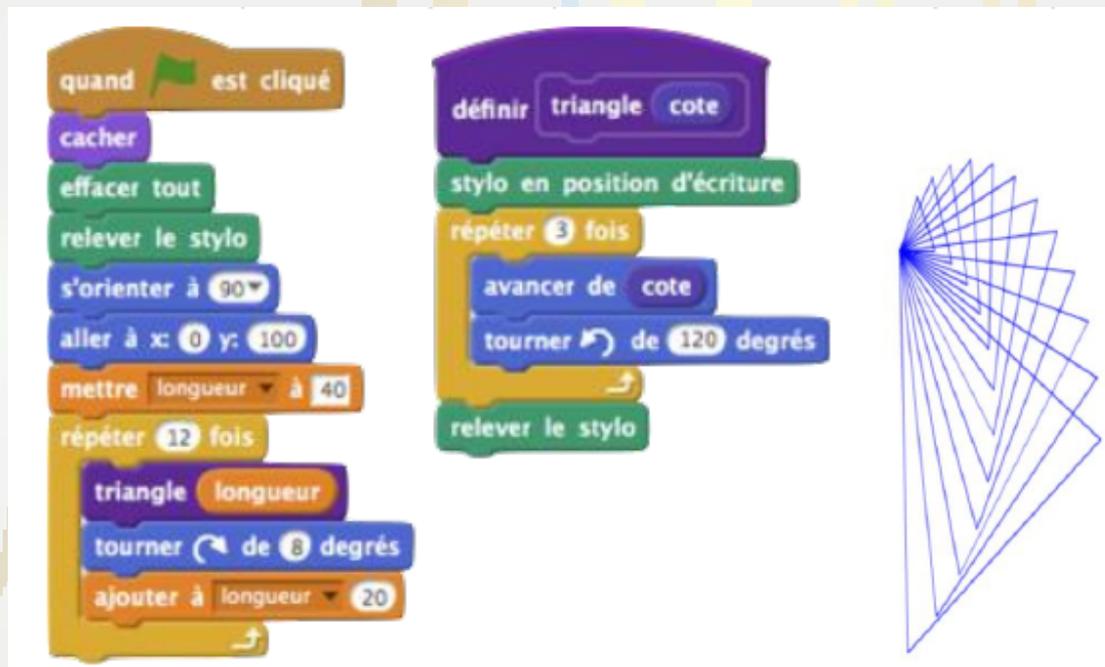


Décomposer un problème avec des fonctions

Fonction Python : valeur renvoyée

Transition Scratch/Python

Changement de technologie (paradigme, fond) ou de technique (forme) ?



```
import turtle

def figure():

    turtle.hideturtle()
    turtle.clear()
    turtle.up()
    turtle.setheading(0)
    turtle.goto(0,100)
    longueur = 40
    for i in range(12):
        triangle(longueur)
        turtle.right(8)
        longueur = longueur + 20

def triangle(cote):
    turtle.down()
    for i in range(3):
        turtle.forward(cote)
        turtle.left(120)
    turtle.up()
```

Transition Scratch/Python

La conjecture de Syracuse

Suite de nombres entiers.

Le successeur d'un nombre N est :

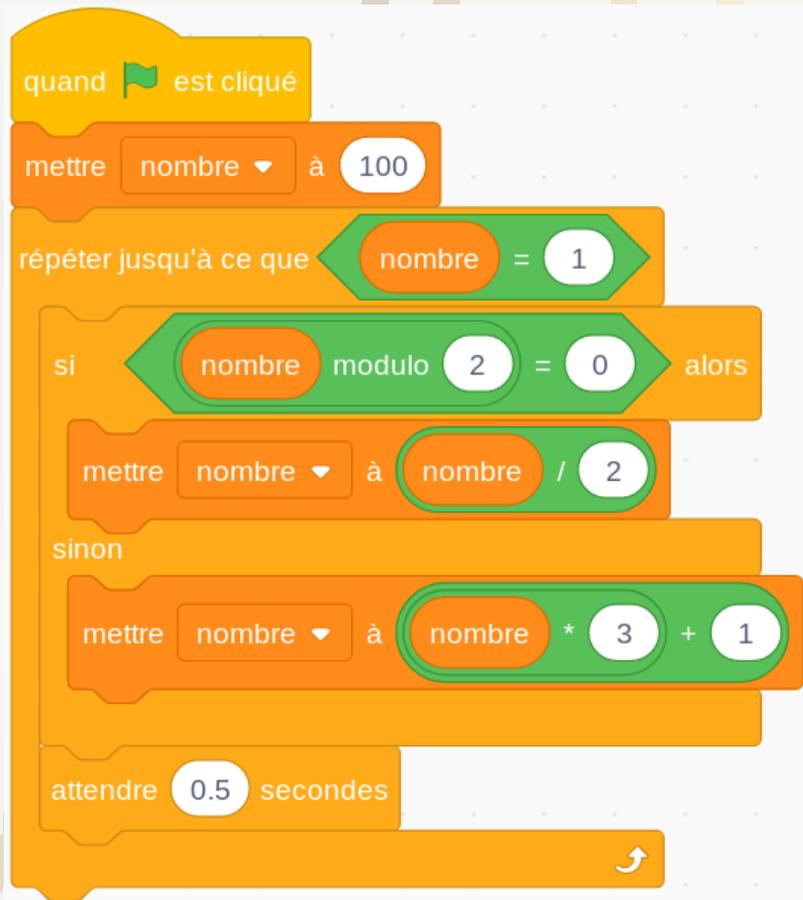
- $N/2$ si N est pair
- $3 * N + 1$ si N est impair

On finit toujours par arriver à 1 (?)

Exemple : 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1
Temps de vol de l'exemple : 13

Transition Scratch/Python

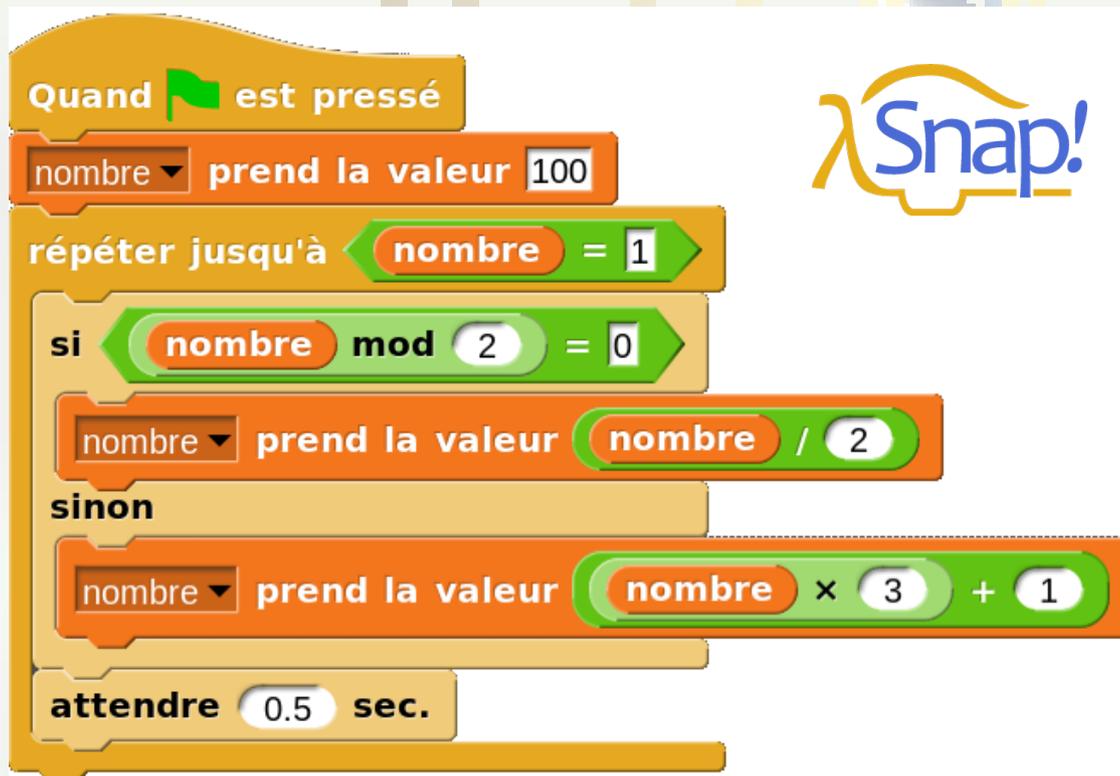
Changement de Technologie



```
nombre=100
while nombre!=1:
    if (nombre%2==0) :
        nombre=nombre/2
    else :
        nombre=nombre*3+1
    print nombre
```

Transition Scratch/Python

Changement de Technologie



The image shows a Scratch script for a number game. The script starts with a 'Quand le drapeau est cliqué' block. It then sets the variable 'nombre' to 100. A 'répéter jusqu'à' loop is used, with the condition 'nombre = 1'. Inside the loop, there is an 'if' block: 'si nombre mod 2 = 0'. If true, the variable 'nombre' is set to 'nombre / 2'. If false ('sinon'), 'nombre' is set to 'nombre * 3 + 1'. After the loop, there is an 'attendre 0.5 sec.' block. The Snap! logo is visible in the background.

```
nombre=100
while nombre!=1:
    if (nombre%2==0) :
        nombre=nombre/2
    else :
        nombre=nombre*3+1
    print nombre
```

Transition Scratch/Python

Changement de Technologie



Quand est pressé

nombre prend la valeur 100

répéter jusqu'à nombre = 1

si est_pair nombre

nombre prend la valeur nombre / 2

sinon

nombre prend la valeur nombre x 3 + 1

attendre 0.5 sec.

+est_pair+ x +

si x mod 2 = 0

rapporte vrai

sinon

rapporte faux

Transition Scratch/Python

Changement de Technologie



```
Quand  est pressé
nombre prend la valeur 100
répéter jusqu'à nombre = 1
nombre prend la valeur suivant nombre
attendre 0.5 sec.
```

```
+est_pair+ x +
si x mod 2 = 0
  rapporte vrai
sinon
  rapporte faux
```

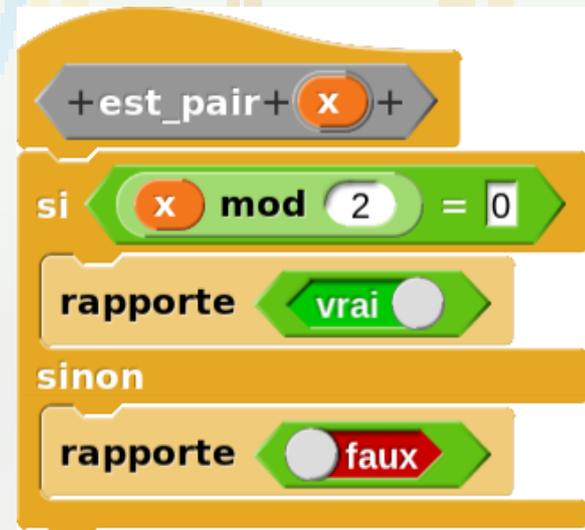
```
+suivant+ x +
si est_pair x
  rapporte x / 2
sinon
  rapporte x x 3 + 1
```

Transition Scratch/Python

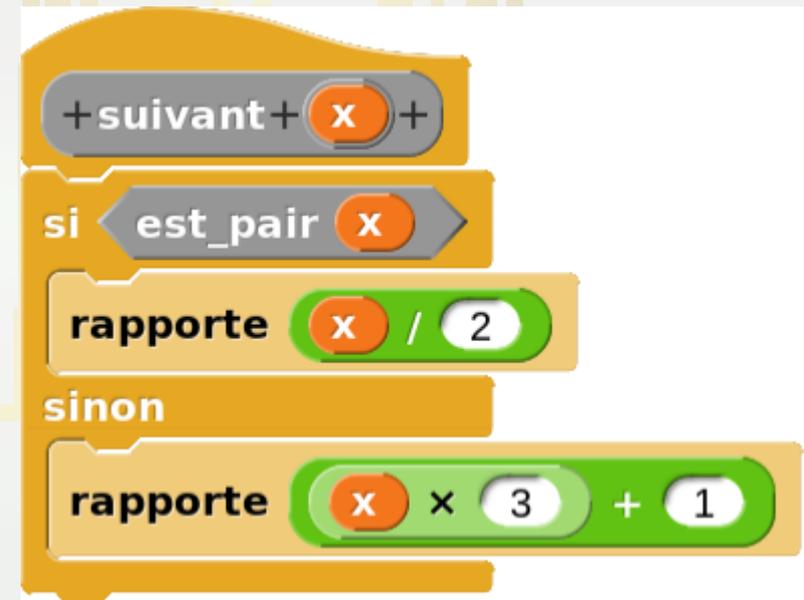
Changement de Technologie



```
+temps+de+vol+ n +  
variables du script longueur  
longueur prend la valeur 0  
répéter jusqu'à n = 1  
n prend la valeur suivant n  
ajouter à longueur 1  
rapporte longueur
```



```
+est_pair+ x +  
si x mod 2 = 0  
rapporte vrai  
sinon  
rapporte faux
```



```
+suivant+ x +  
si est_pair x  
rapporte x / 2  
sinon  
rapporte x x 3 + 1
```



```
temps de vol 500  
110
```

Transition Scratch/Python

Changement de Technologie

The Scratch script on the left calculates the average of numbers from 1 to n. It uses a loop to sum the numbers and count them, then reports the average.

```
+temps+ moyen+ de+ vol+ des+ nombres+ jusqu+ à+ n +  
variables du script x somme  
x prend la valeur 1  
somme prend la valeur 0  
répéter n fois  
ajouter à somme temps de vol x  
ajouter à x 1  
rapporte somme / n
```

The Scratch script on the right calculates the length of numbers from 1 to n. It uses a loop to count the numbers and reports the total count.

```
+temps+ de+ vol+ n +  
variables du script longueur  
longueur prend la valeur 0  
répéter jusqu'à n = 1  
n prend la valeur suivant n  
ajouter à longueur 1  
rapporte longueur
```

21.32

temps moyen de vol des nombres jusqu'à 50

Transition Scratch/Python

Changement de Technologie

A travers cet exemple, on peut voir :

- Comment on décompose le problème en fonctions
- L'utilisation des blocs construits peut s'utiliser un peu comme en mode « console »
- Comment l'impossibilité d'écrire des fonctions (qui renvoient une valeur) amène à un affichage qui devient inopportun dès qu'on utilise des fonctions qui renvoient une valeur.
- La création de fonctions permet de masquer la complexité de l'algorithme et permet de développer un algorithme plus complexe. Attention à la mise à disposition systématique de fonctions que l'élève n'écrit pas (principe des bibliothèques dans les langages) : l'élève peut ne pas percevoir l'intérêt de la création de fonctions pour soulager la charge cognitive. En lui permettant de fabriquer lui-même ce qui lui permet d'alléger la complexité, il a à la fois une vue d'ensemble et à la fois une compréhension des détails

Transition Scratch/Python

Expérimentation

➤ Deux classes de troisième (3D classe expérimentale et 3E classe témoin)

Depuis le mois de mars :

3D : utilisent Snap! Créent des blocs qui « rapportent » des valeurs et des blocs « assertion »

3E : utilisent Scratch. Ils ont les mêmes situations que les 3D

A partir de juin :

Les deux classes utilisent Python

Observation : compréhension et utilisation des fonctions en Python

Transition Scratch/Python

Expérimentation

3D (Snap!)

Eve

```
Quand le drapeau est cliqué  
  nombre prend la valeur 29  
  répéter 40 fois  
    si nombre EST PAIR  
      nombre prend la valeur nombre / 2  
    sinon  
      nombre prend la valeur nombre x 3 + 1  
    attendre 1 sec.
```

The image shows a Scratch script starting with a 'When green flag is clicked' event. It sets a variable 'nombre' to 29. A loop repeats 40 times. Inside the loop, an 'if' block checks if 'nombre' is even. If true, it divides 'nombre' by 2. If false, it multiplies 'nombre' by 3 and adds 1. Finally, it waits for 1 second.

```
+ X + EST + PAIR +  
si X mod 2 = 0  
  rapporte vrai  
sinon  
  rapporte faux
```

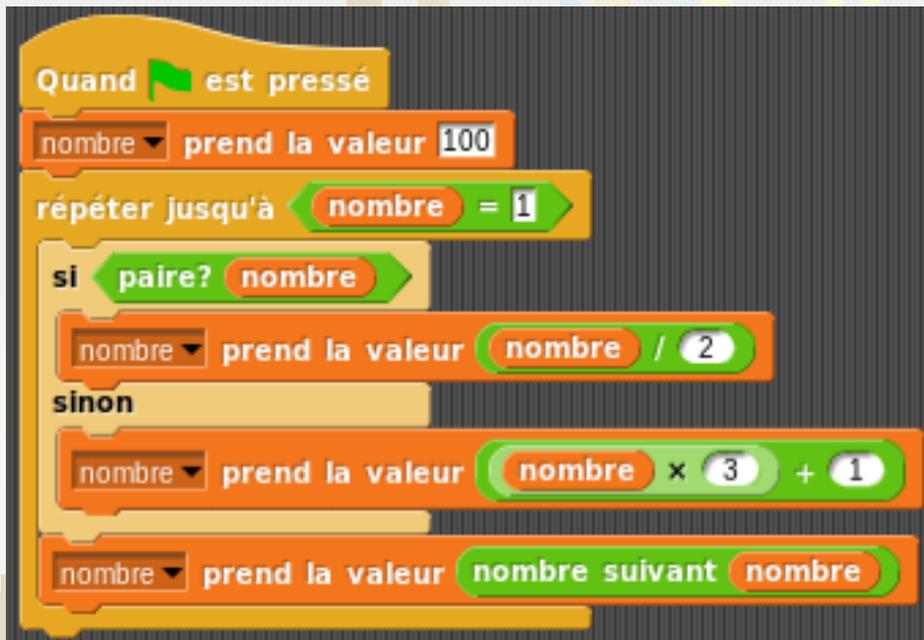
The image shows a Python code snippet. It starts with a comment '+ X + EST + PAIR +'. Then it has an 'if' statement 'si X mod 2 = 0'. If true, it prints 'rapporte vrai'. If false, it prints 'rapporte faux'.

Transition Scratch/Python

Expérimentation

3D (Snap!)

Lara



```
Quand le drapeau vert est cliqué
  nombre prend la valeur 100
  répéter jusqu'à nombre = 1
  si paire? nombre
    nombre prend la valeur nombre / 2
  sinon
    nombre prend la valeur nombre * 3 + 1
  nombre prend la valeur nombre suivant nombre
```

The image shows a Scratch script for the Collatz conjecture. It starts with a 'When green flag is clicked' event block. The first block is 'number takes the value 100'. This is followed by a 'repeat until' loop with the condition 'number = 1'. Inside the loop, there is an 'if-else' block. The 'if' part checks 'is number even?'. If true, it sets 'number' to 'number / 2'. If false, it sets 'number' to 'number * 3 + 1'. After the if-else block, there is a 'number takes the value next number' block, which increments the number by 1.

Transition Scratch/Python

Expérimentation

3E (Scratch)

Andréa



```
quand est cliqué
mettre temps de vol à 0
mettre nombre à 0
mettre nombre à 23
répéter jusqu'à ce que nombre = 1
  si nombre modulo 2 = 0 alors
    mettre nombre à nombre / 2
  sinon
    mettre nombre à nombre * 3 + 1
ajouter 1 à temps de vol
dire temps de vol pendant 2 secondes
```

The image shows a Scratch script on a grid background. The script starts with a yellow 'when green flag is clicked' block. It then sets three variables: 'temps de vol' to 0, 'nombre' to 0, and 'nombre' to 23. A 'repeat until' loop is set to 'nombre = 1'. Inside the loop, there is an 'if' block: 'if nombre modulo 2 = 0 then' followed by 'set nombre to nombre / 2', and 'else' followed by 'set nombre to nombre * 3 + 1'. After the loop, there is an 'add 1 to temps de vol' block and a 'say temps de vol for 2 seconds' block.

Transition Scratch/Python

Expérimentation

3E (Scratch)

William



```
quand [drapeau] est cliqué
mettre [NOMBRE DE BASE] à 1
mettre [temps de vol] à 0
répéter 100 fois
  mettre [nombre] à [NOMBRE DE BASE]
  répéter jusqu'à ce que [nombre = 1]
    si [nombre modulo 2 = 0] alors
      mettre [nombre] à [nombre / 2]
    sinon
      mettre [nombre] à [nombre * 3 + 1]
  ajouter 1 à [temps de vol]
  ajouter 1 à [NOMBRE DE BASE]
mettre [temps de vol] à [temps de vol / 100]
```

(Pour le temps de vol moyen des nombres jusqu'à 100) :
« Est-ce que je peux prendre des valeurs au hasard entre 1 et 100 ? »

Transition Scratch/Python

Expérimentation

3E (Scratch)

Vanessa



Calcul du temps de vol :

J'ai enlevé "attendre 1 seconde" parce que ça donne directement le nombre d'étape.

Calcul du temps de vol moyen :

Il y avait une autre difficulté c'était d'additionner toutes les étapes de chaque nombre de 1 à 100.

Je sais ce que je veux faire mais je n'arrive pas à le mettre en programme. Je n'arrive plus à l'avoir en entier dans ma tête.