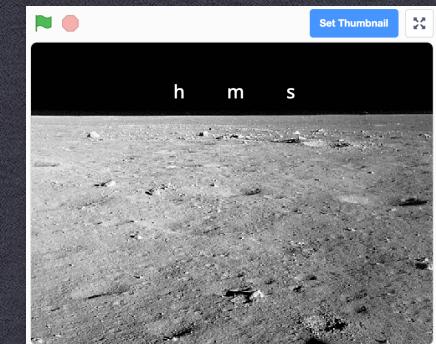




[scratch.mit.edu](https://scratch.mit.edu)



[@SamChaTICE](#)



[Studio Scratch de départ](#)

PROJET

# FICHES SCRATCH - HORLOGES

## LE TEMPS QUI PASSE

DATE

NOVEMBRE 2025

CIBLE

ATELIER CANOPÉ 88



Samuel Chalifour



**Créez les variables *secondes*, *minutes* et *heures*.**

**Faites en sorte que le Drapeau vert initialise ces variables à 0. Puis les secondes doivent défiler très vite indéfiniment.**

Événements

Contrôle

Variables

mettre **heures** ▾ à

répéter indéfiniment

mettre **secondes** ▾ à

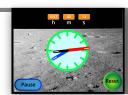
mettre **minutes** ▾ à

ajouter à **secondes** ▾

quand  est cliqué



Arrière-plans



**L'ordre des blocs d'initialisation n'a pas d'importance.  
Placez bien les étiquettes des variables au-dessus de  
leur nom sur la scène.**

Événements      Contrôle      Variables

The Scratch script consists of the following blocks:

- Event: **quand [drapeau] est cliqué**
- Control: **mettre [secondes v] à [0]**
- Control: **mettre [minutes v] à [0]**
- Control: **mettre [heures v] à [0]**
- Control: **répéter indéfiniment**
  - Control: **ajouter [1] à [secondes v]**

Arrière-plans



Pour faire avancer les *minutes*, on fait un test *si...alors*. Et on remet les *secondes* à 0. Ce script doit être ajouté au bon endroit dans le script précédent. Maintenant, au *Drapeau vert*, les minutes défilent, moins vite que les secondes.

Contrôle

Opérateurs

Variables



Arrière-plans

secondes

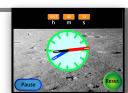
si      alors

=

attendre 1 secondes

mettre secondes à

ajouter 1 à minutes



**L'important est de faire attention à la condition du si... alors, ici (secondes=60), et à l'ordre des blocs. Il existe plusieurs solutions, à vous de juger des différences.**

Contrôle

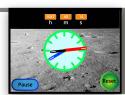
Opérateurs

Variables

```
mettre heures à 0
répéter indéfiniment
  ajouter 1 à secondes
  si secondes = 60 alors
    mettre secondes à 0
    ajouter 1 à minutes
  fin
fin
```



Arrière-plans



De la même façon, pour faire avancer les *heures*, on fait le même test. Et on remet à zéro les *minutes*. Ce morceau de script doit être ajouté au bon endroit dans le script précédent. On ajoute une instruction d'attente pour être plus au rythme des vraies secondes. On ne doit jamais voir le chiffre 60 s'afficher.

Contrôle

Opérateurs

Variables



Arrière-plans

si      alors

=

mettre minutes à

minutes

ajouter à heures

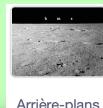


Il s'agit surtout de mettre l'instruction d'attente au bon endroit. Vous devrez adapter la valeur de cette attente pour coller au plus près du bon rythme d'une horloge.

Contrôle

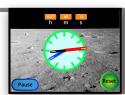
Opérateurs

Variables



Arrière-plans

```
mettre heures à 0
répéter indéfiniment
  attendre 1 secondes
  ajouter 1 à secondes
  si secondes = 60 alors
    mettre secondes à 0
    ajouter 1 à minutes
    si minutes = 60 alors
      mettre minutes à 0
      ajouter 1 à heures
  fin si
fin répétition
```



Faites apparaître le bouton *Reset* qui remet tout à zéro quand on clique dessus. On agit sur son apparence (changement de costume) pour faire croire que le bouton est vraiment appuyé. Puis on le remet en normal une fois qu'il a fait son effet.

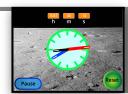
Apparence

Événements

Contrôle

Variables





**Rien de particulier ici, veillez à bien placer l'attente entre les deux changements de costume, sinon les calculs des autres blocs sont trop rapides pour être vus à l'oeil humain, l'ordre n'a donc pas d'importance.**

Apparence

Événements

Contrôle

Variables

quand ce sprite est cliqué

costume suivant

mettre secondes à 0

mettre minutes à 0

mettre heures à 0

attendre 0.1 secondes

costume suivant



Reset



Faites apparaître le bouton Start. Au Drapeau vert, il passe en costume Pause. Quand on clique dessus, s'il est sur Pause, il repasse sur le costume Start et arrête tout, sinon, il repasse en Pause et envoie le message start. Le script des Arrières-plans doit être séparé en 2 scripts et les nouveaux blocs doivent leur être associés.

Apparence      Événements      Contrôle      Opérateurs

costume suivant      costume suivant

envoyer à tous start

si alors  
sinon stop tout

= Pause

quand ce sprite est cliqué nom du costume

quand vert est cliqué basculer sur le costume Pause

Start

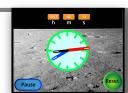
quand je reçois start

quand vert est cliqué

mettre secondes à 0  
mettre minutes à 0  
mettre heures à 0  
répéter indéfiniment ajouter 1 à secondes  
si secondes = 60 alors mettre secondes à 0 ajouter 1 à minutes  
si minutes = 60 alors mettre minutes à 0 ajouter 1 à heures

envoyer à tous start

Arrière-plans



**Le bloc *stop tout*, arrête vraiment l'exécution de tous les scripts.**  
**Saurez-vous refaire ce script de *Start* en mieux ? Vous pouvez enlever un bloc *Costume suivant* et utiliser un *si...alors*, plutôt qu'un *si...alors...sinon*.**

Apparence      Événements      Contrôle      Opérateurs

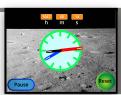
Start

Arrière-plans

The script consists of two main sections. The left section, triggered by a green flag, sets a timer and sends a 'start' message to all other sprites. The right section, triggered by a blue start button, contains a loop that adds 1 second to the timer every 60 seconds, and sends a 'start' message to all other sprites when the timer reaches 60 seconds.

```
when green flag is clicked
  set [seconds v] to [0]
  set [minutes v] to [0]
  set [hours v] to [0]
  send start to everyone
repeat (60)
  add (1) to [seconds v]
  if [seconds v] = [60] then
    set [minutes v] to [0]
    add (1) to [hours v]
    send start to everyone
  end
end

when start button is pressed
  stop [all v]
  switch costume to [1]
  send start to everyone
```



Activez le deuxième arrière-plan de la Scène. Montrez le lutin *Aiguille secondes*. Au *Drapeau vert*, celle-ci doit tourner de 6° toutes les secondes (un tour complet en 360°). À chaque fois qu'elle revient à son point de départ, elle envoie un message pour prévenir qu'on doit ajouter une minute.

## Opérateurs

quand  est cliqué

aller à x: 0 y: 0

s'orienter à 90

attendre 1 secondes

$$\text{ } = 90$$

Aiguille se...

quand je reçois start ▾

quand je reçois **reset** ▼

s'orienter à 90

si  alors

[envoyer à tous](#) [newMinute ▾](#)

répéter indéfiniment

tourner  de 6 degrés

## direction



Les boutons Reset et Start doivent aussi fonctionner pour cette aiguille. C'est elle qui va rythmer le déplacement des deux autres aiguilles. Au lieu de calculer soi-même la rotation de  $6^\circ$ , on aurait pu faire le calcul par des blocs :  $360$  (un tour complet) divisé par  $60$  (nombre de secondes dans une minutes) :

$$360 / 6$$

Mouvement

Événements

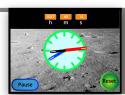
Contrôle

Opérateurs

The Scratch script illustrates the logic for the needle's movement. It consists of three main parts:

- When green flag clicked:** Sets the needle to position (0, 0) and orientation 90 degrees.
- When I receive "reset":** Sets the needle to orientation 90 degrees.
- When I receive "start":** The script enters a loop that repeats indefinitely. Inside the loop, it waits 1 second and then turns the needle 6 degrees clockwise. It then checks if the needle's direction is 90 degrees. If so, it sends a "newMinute" message to all other turtles.

A small icon in the top right corner shows a red needle with the text "Aiguille se...".



Faites la même chose pour l'Aiguille minutes, mais elle n'avance qu'en recevant le message *newMinute*. Quand elle fait un tour complet, elle envoie le message *newHeure*. Trouvez, avec les mêmes blocs, comment doit être programmée l'Aiguille heures.

Mouvement

Événements

Contrôle

Opérateurs

quand  est cliqué

aller à x: 0 y: 0

s'orienter à 90

quand je reçois reset

s'orienter à 90

tourner  de 6 degrés

envoyer à tous newheure



Aiguille ...

Aiguille h...

= 90

quand je reçois newminute

direction

si alors



Pour l'Aiguille heures, c'est donc bien plus simple, sauf si vous voulez rajouter le compteur de jours, voire d'années... mais cela deviendra plus complexe avec les années bissextiles du fait que les rotations de la Terre durent en fait 23h56min4sec soit un peu de moins de 24h.

Trouvez d'abord un moyen de mieux rythmer l'horloge puisqu'il y a en fait un décalage dû au fait qu'il y a plus d'une seconde entre chaque attente d'une seconde. Cherchez pas exemple dans les blocs « Capteurs ».

Mouvement

Événements

Contrôle

Opérateurs



quand je reçois newminute ▾

tourner ⚡ de 6 degrés

si direction = 90 alors

envoyer à tous newheure ▾



quand je reçois newheure ▾

tourner ⚡ de 6 degrés