

ALLARD Charlotte  
FIAT Xavier  
MEISSIREL Elise  
RAPEGNO Virgile

# Coding weeks : semaine 2

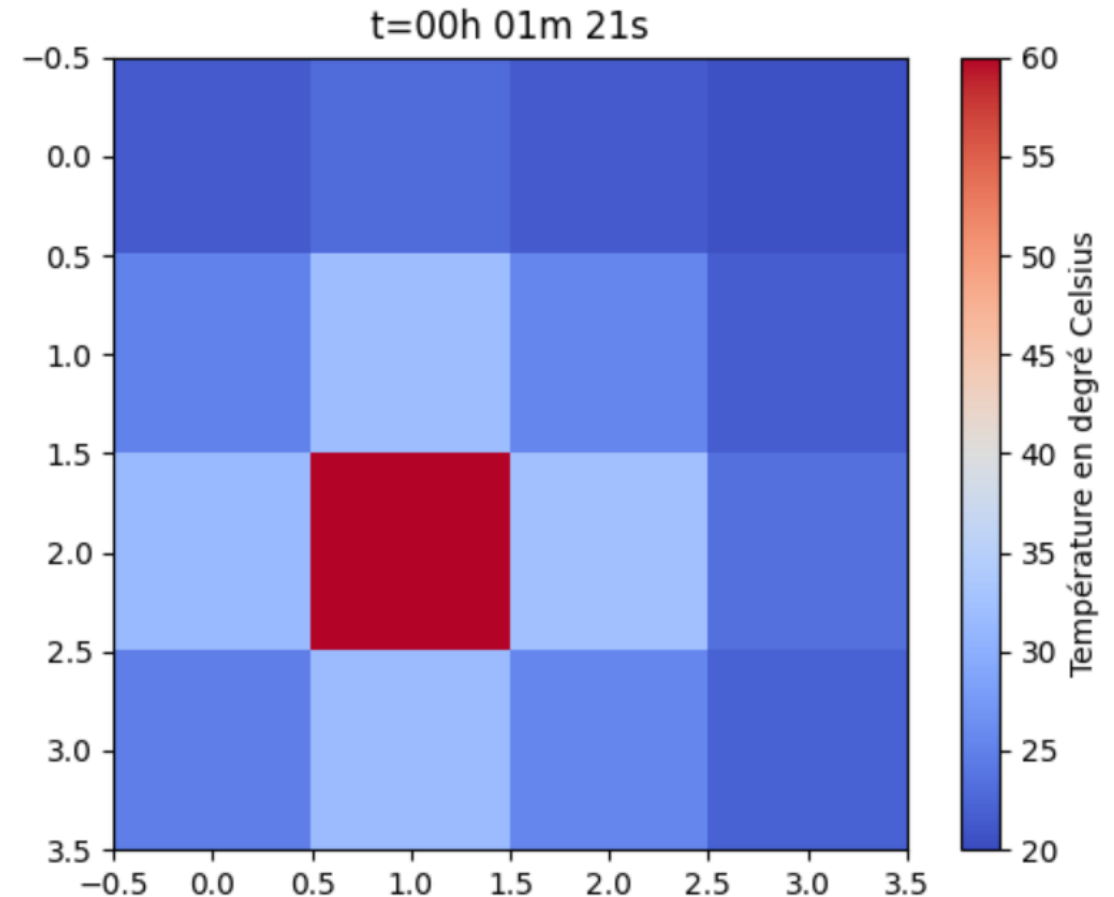


Application de l'automate cellulaire au transfert thermique

*Dépôt GitLab : <https://gitlab-ovh-04.cloud.centralesupelec.fr/virgile.rapegno/cw-semaine-2-xcev>*

# Présentation du projet

- L'univers représente une tranche d'un matériau supposé infiniment long.
- Chaque case contient la température de la portion du matériau correspondante.
- A l'aide des équations régissant les transferts thermiques (loi de Newton et équation de la chaleur), on modifie les cellules en fonction de leur entourage.



# Intérêts du projet

---

- **Usage pédagogique :**

L'animation proposée peut servir aux élèves pour visualiser les phénomènes de transfert thermique et ainsi mieux les comprendre

- **Usage industriel :**

En première approximation, le projet peut servir à connaître la température d'un matériau en fonction des conditions dans lesquelles il évolue. On peut par exemple trouver à quel instant il n'est plus dangereux de le toucher, quand est-ce qu'il atteint sa température de fusion...

# Répartition en sprints

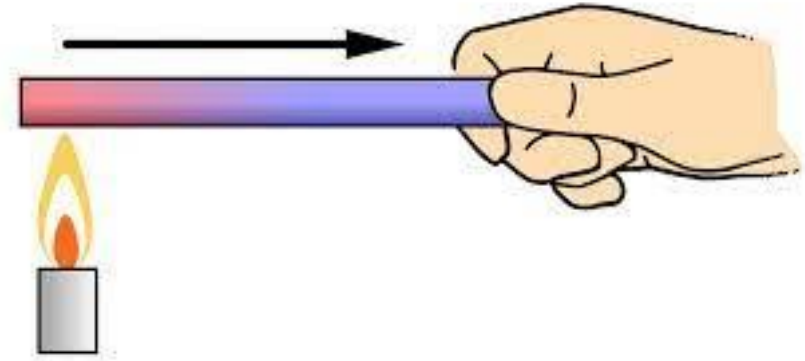
---

## **Sprint 1 : Initialisation**

- Création d'un univers vide
- Placer une source
- Placer les sources
- Créer un dictionnaire contenant les données thermodynamiques de plusieurs matériaux

## **Sprint 2 : Evolution de l'univers**

- Faire évoluer une cellule selon son voisinage
- Faire évoluer l'univers dans son ensemble



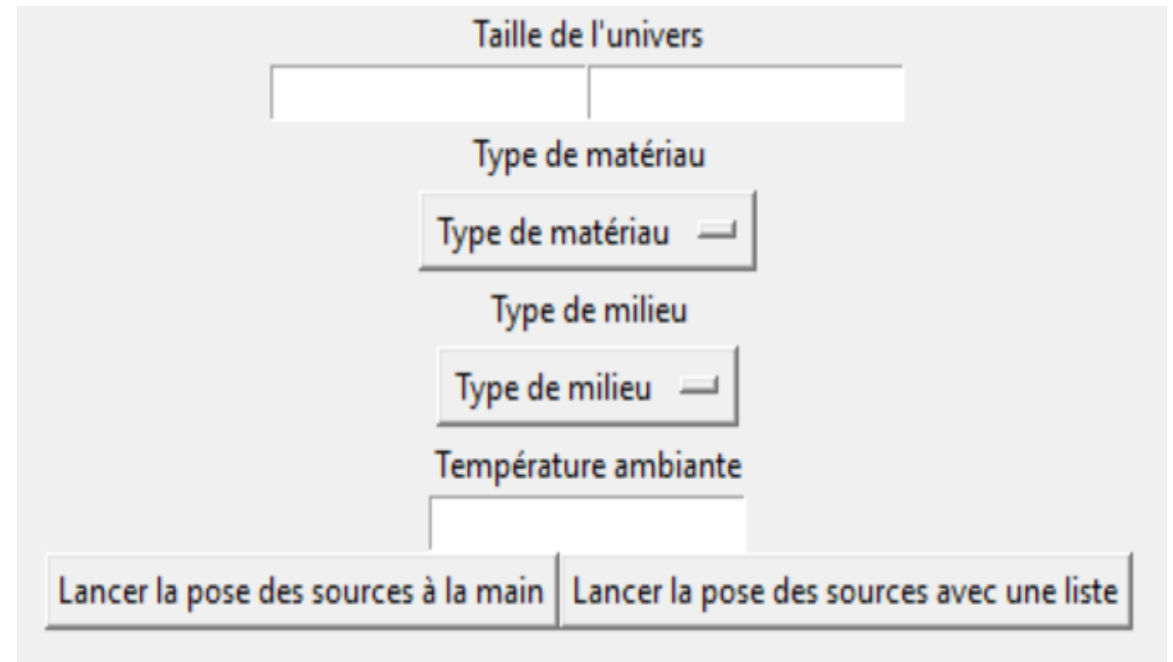
# Répartition en sprints

---

## Sprint 3 : Interface graphique

- Afficher l'univers en ajoutant des couleurs pour représenter le chaud et le froid (avec matplotlib)
- Créer l'interface utilisateur pour permettre à l'utilisateur d'entrer la taille de l'univers, choisir le matériau, placer les sources de chaleur... (avec tkinter)

## Obtention d'un MVP



Taille de l'univers

Type de matériau

Type de milieu

Température ambiante

Lancer la pose des sources à la main

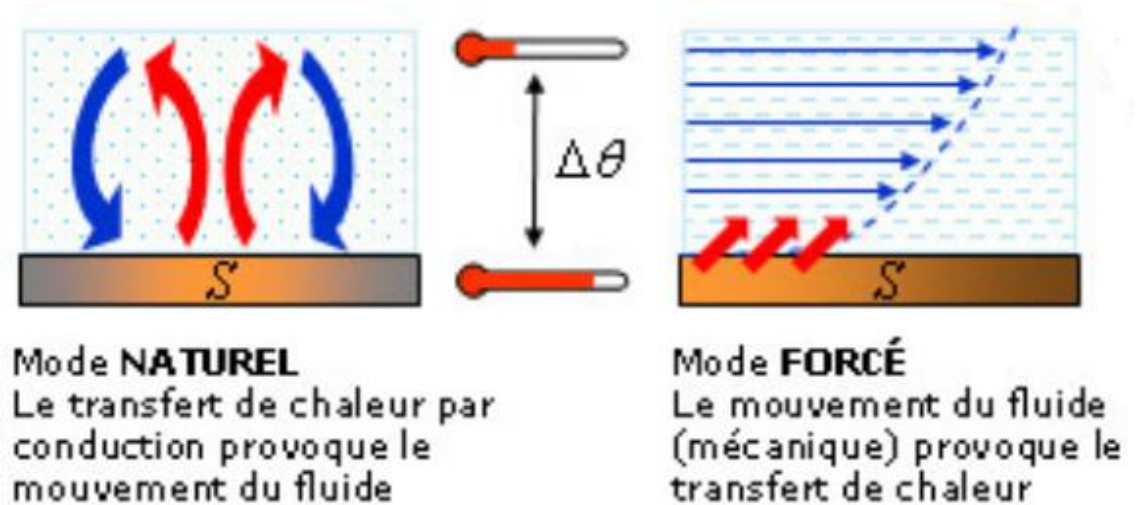
Lancer la pose des sources avec une liste

# Répartition en sprints

---

## Sprint 4 : Prise en compte de la conducto-convection

- Créer un dictionnaire contenant le coefficient conducto-convectif de plusieurs milieux
- Prendre en compte la conducto-convection lors de l'évolution de l'univers



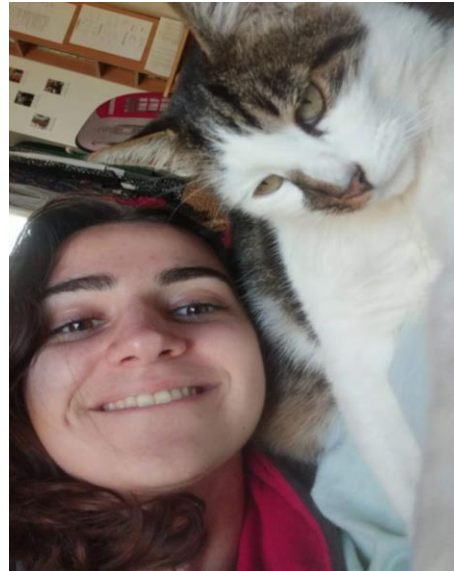
# Notre rôle au sein de l'équipe

---



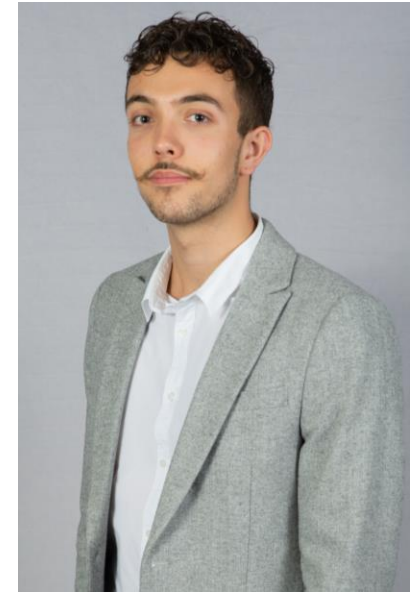
Charlotte & Xavier

Responsables physique et évolution de l'univers



Elise

Responsable matplotlib et aide autres fonctionnalités



Virgile

Responsable interface utilisateur et qualité du code

# Partie physique

---

On s'appuie sur l'équation de la chaleur pour représenter la diffusion de la chaleur d'une cellule à l'autre :  $\frac{\partial T}{\partial t} = D \Delta T$

On discrétise l'équation par la méthode d'Euler :

$$\begin{aligned} T(x, y, nt) \\ = T(x, y, (n-1)t) + \Delta t D * \left[ \frac{T(x+1, y, (n-1)t) + T(x-1, y, (n-1)t) - 2T(x, y, (n-1)t)}{(\Delta x)^2} + \frac{T(x, y+1, (n-1)t) + T(x, y-1, (n-1)t) - 2T(x, y, (n-1)t)}{(\Delta y)^2} \right] \end{aligned}$$



# Partie physique



La démarche est la même sur les bords, où les transferts sont régis par la loi de Newton :

$$\vec{j}_{th} = h * [T_{solide} - T_{fluide}] \vec{n}$$

On en déduit, par la loi de Fourier et la continuité du vecteur densité de flux thermique à l'interface du matériau :

$$T(x, y_{bord}, n * t) = \frac{\lambda}{\lambda + h * \Delta y} * T(x, y_{bord} - 1, n * t) + \frac{h}{h + \frac{\lambda}{\Delta y}} * T_{fluide}$$