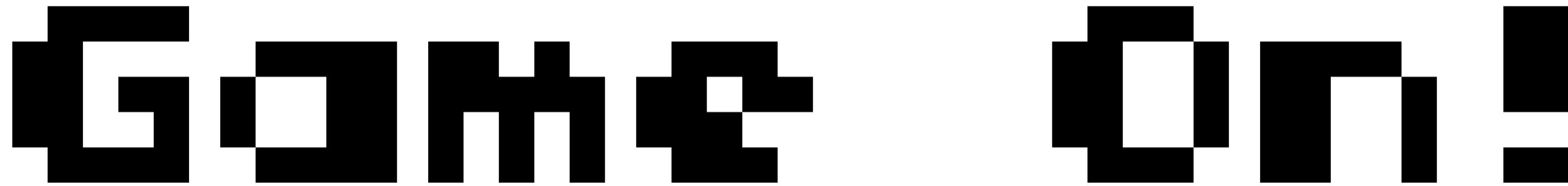




STAGE 1





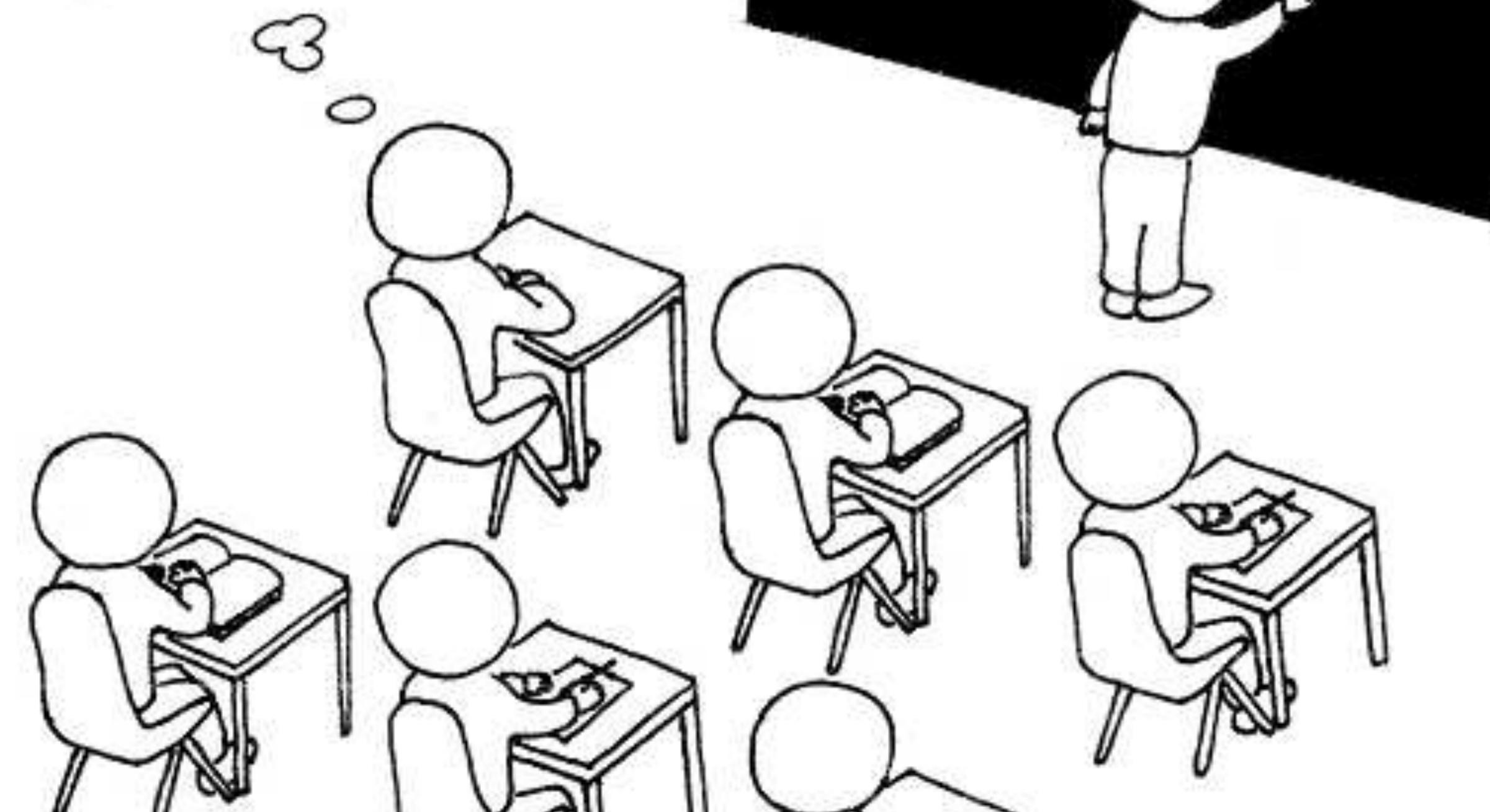
# **Initiation à la programmation par le Rétrogaming**

**M. Liedloff & N. Ollinger**  
**UFR Sciences & Techniques, UEO 2026**

THUS, FOR ANY NONDETERMINISTIC TURING MACHINE  $M$  THAT RUNS IN SOME POLYNOMIAL TIME  $p(n)$ , WE CAN DEVISE AN ALGORITHM THAT TAKES AN INPUT  $\omega$  OF LENGTH  $n$  AND PRODUCES  $E_{n,\omega}$ . THE RUNNING TIME IS  $O(p^2(n))$  ON A MULTITAPE DETERMINISTIC TURING MACHINE AND...

WTF, MAN. I JUST  
WANTED TO LEARN  
HOW TO PROGRAM  
VIDEO GAMES.

SIPSER CH7  
 $y_{i,j-1} \wedge y_{i,j} \wedge y_{i,j+1} \wedge y_{i,j+2}$   
 $y_{i,j-1} \wedge y_{i,j} \wedge y_{i,j+1} \wedge y_{i,j+2}$   
 $N_i = (A_{i,0} \vee B_{i,0}) \wedge (A_{i,1} \vee B_{i,1}) \wedge \dots \wedge$   
 $N = N_0 \wedge N_1 \wedge \dots \wedge N_n$



# Organisation

## Administrivia

**Cours** 1h M. Liedloff & N. Ollinger (salle E12)

**TD** 7x 2h M. Liedloff & N. Ollinger (salles E18 & E19)

## Support de cours

<https://celene.univ-orleans.fr/course/view.php?id=13657>

## Évaluation

soutenance (50%) + rapport à rendre (50%)

**Objectif** Découvrir la programmation Python à travers la conception, en groupe, d'un jeu vidéo simple pour une console imaginaire (*fantasy console*) d'inspiration rétrogaming.

# **Évaluation :** **Game Jam**

# Game Jam

## modalités d'évaluation

L'objectif de cette UEO est de créer des jeux vidéos.

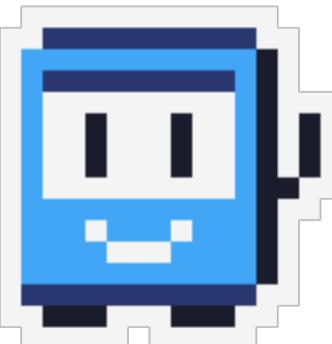
L'évaluation se fera sur **vos productions** !

Organisation d'une Game Jam dont les thèmes seront dévoilés un peu plus tard dans le semestre.

Travail en **groupe de 3 étudiants** : dans un premier temps, répondre rapidement au **sondage sur Celene**.

Le **13 avril matin** : présentation des jeux produits, mise en ligne, vote du public, remise du rapport !

# À la découverte de TIC-ED



# **TIC-80 : spécifications**

**La fantasy console qu'il vous faut !**

**AFFICHAGE** un écran de 240x136 pixels en 16 couleurs.

**ENTRÉES** 4 gamepads / souris / clavier.

**SPRITES** 256 sprites en 8x8 + 256 tuiles en 8x8.

**CARTE** 240x136 tuiles, soit 1920x1088 pixels.

**SON** 4 canaux avec forme d'onde configurable.

**CODE** jusqu'à 64Kio en Lua / ... / **Python** / ...

TIC-80 tiny computer  
version 1.1.2837 (be42d6f)  
<https://tic80.com> (C) 2017-2023

Hello! type help for help

> 



(0,0)

(239,0)

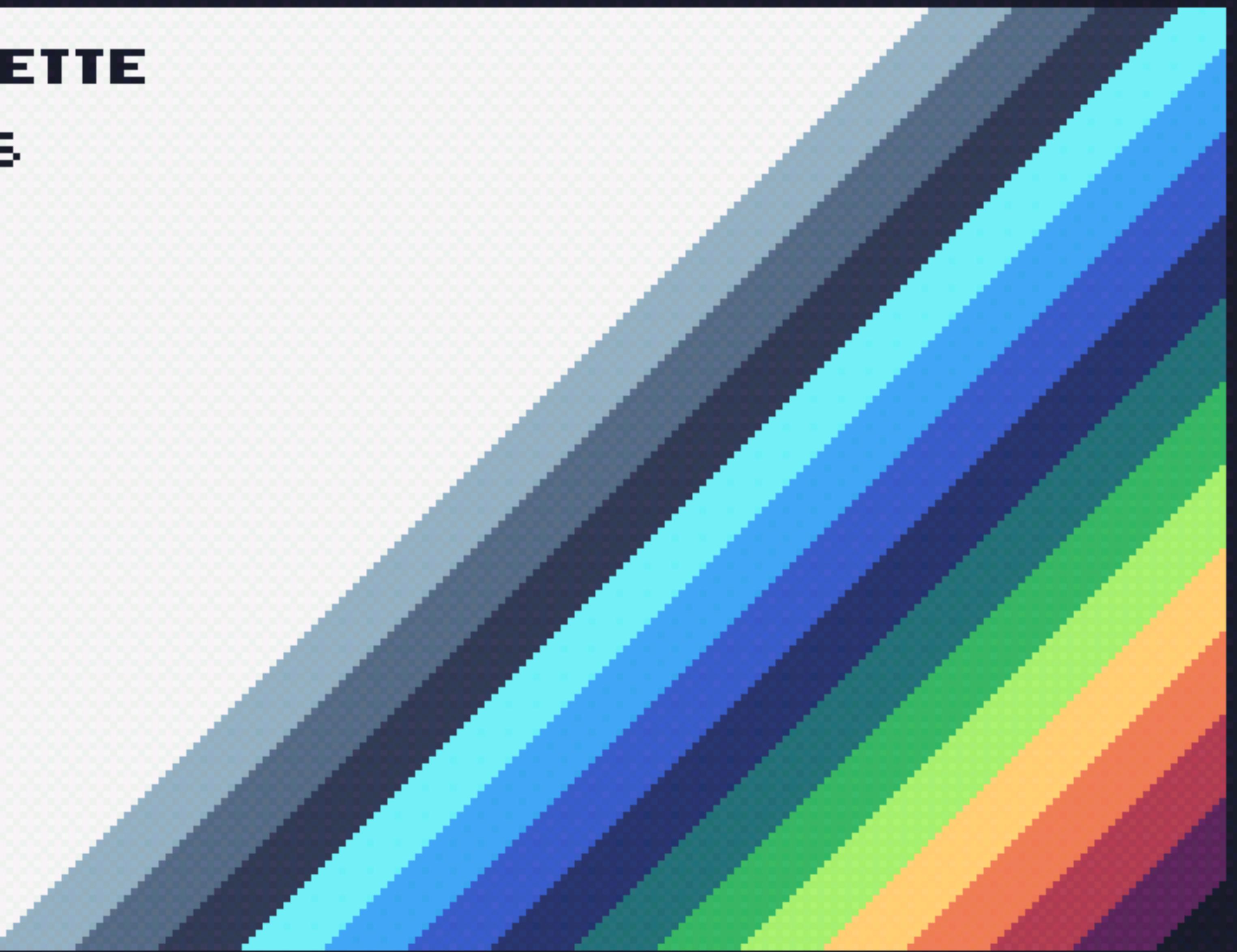
(x,y)

coordenadas TIC-80

(0,135)

## **SELECT PALETTE**

- > **SWEETIE-16**
- DB16
- PIC0-8
- ARNE16
- EDG16
- A64
- C64
- VIC20
- CGA
- SLIFE
- JMP
- CGARNE
- PSYVG
- EROGE
- EISLAND



# Cartouches de jeu TIC-80

les fichiers .tic

Un jeu = un projet = un fichier .tic

Quelques Kio seulement, facile à stocker, facile à échanger !

Tous les jeux TIC-80 sont **modifiables** facilement.

Possibilité d'**exporter** pour le web ou vers les formats natifs courants (Windows, macOS, Linux).



# **Programmer en Python**





English

Español

தமிழ்

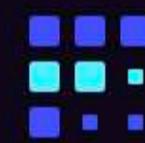
Github

# futurecoder

Le code du futur, c'est vous !



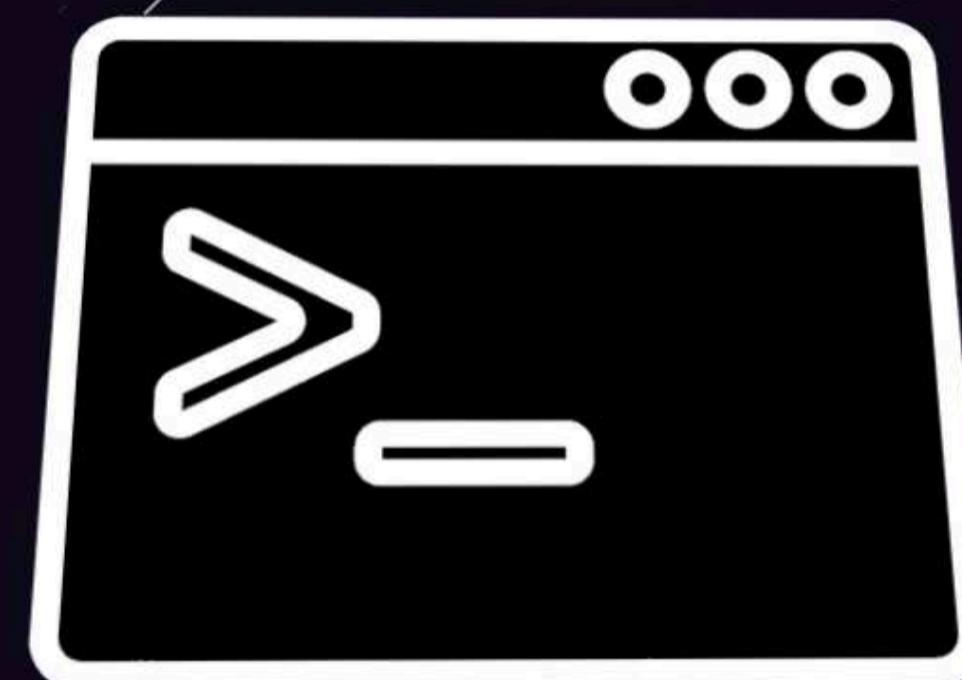
Du nouveau dans les cours en ligne interactifs



100 % Libre et Open-Source



Spécialement adapté si vous débutez en Python



>>> Débuter le cours

ou

>>> Utiliser Python

```
1 # script:  python
2 from random import randint
3
4 cls()
5 for x in range(17):
6     for y in range(13):
7         pix(10*x, 10*y, 3)
8
9 t=0
10 col=1
11 x,y=0,0
12
13 def TIC():
14     global t,x,y,col
15     t=t+1
16     if t%5==0:
17         dx=randint(0,1)
18         dy=1-dx
19         x_dest=x+10*dx
20         y_dest=y+10*dy
21
22         if x<160 and y<120:
23             line(x,y,x_dest,y_dest,col)
24             x,y=x_dest,y_dest
25         else:
26             x,y=0,0
27             col=(col*15)+1
```

```
1 # script:  python
2 from random import randint
3
4 cls(0)
5 for x in range(17):
6     for y in range(13):
7         pix(10*x, 10*y, 3)
8
9 t=0
10 col=1
11 x,y=0,0
12
13 def TIC():
14     global t, x, y, col
15     t=t+1
```

```
13 def TICK():
14     global t, x, y, col
15     t=t+1
16     if t%5==0:
17         dx=randint(0, 1)
18         dy=1-dx
19         x_dest=x+10*dx
20         y_dest=y+10*dy
21
22     if x<160 and y<120:
23         Line(x, y, x_dest, y_dest, co
24         x, y=x_dest, y_dest
25     else:
26         x, y=0, 0
27         col=(col%15)+1
```