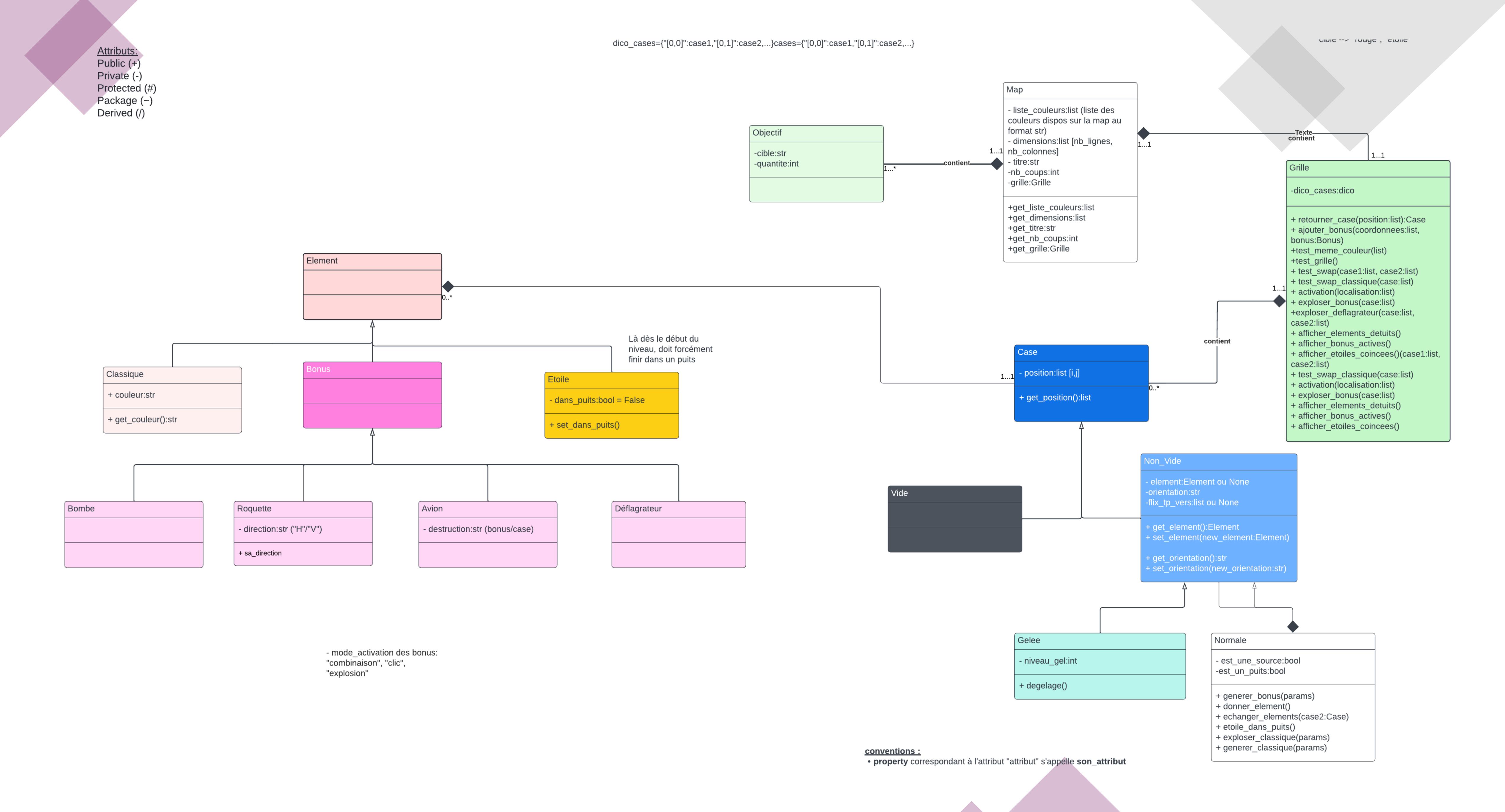
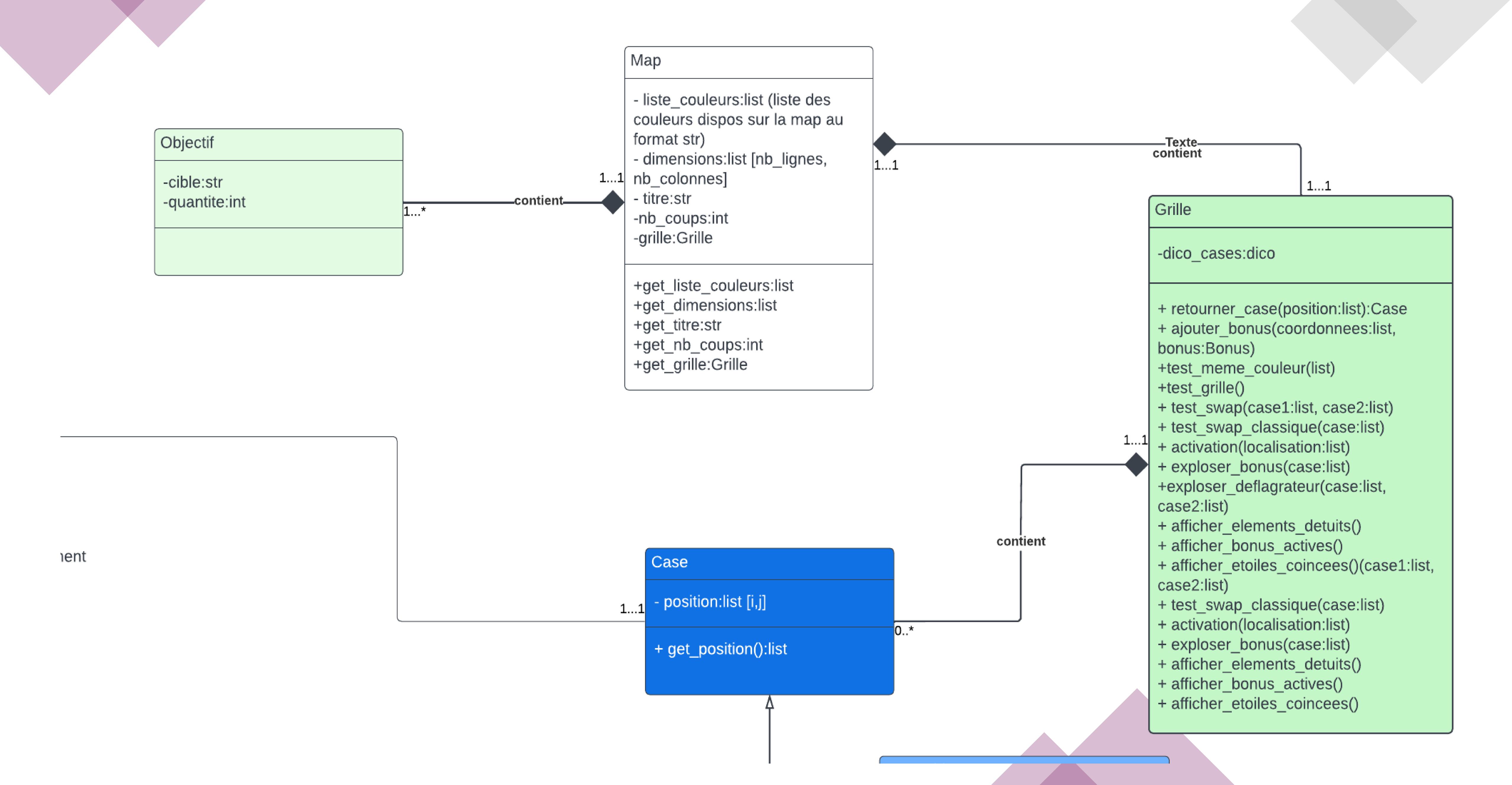


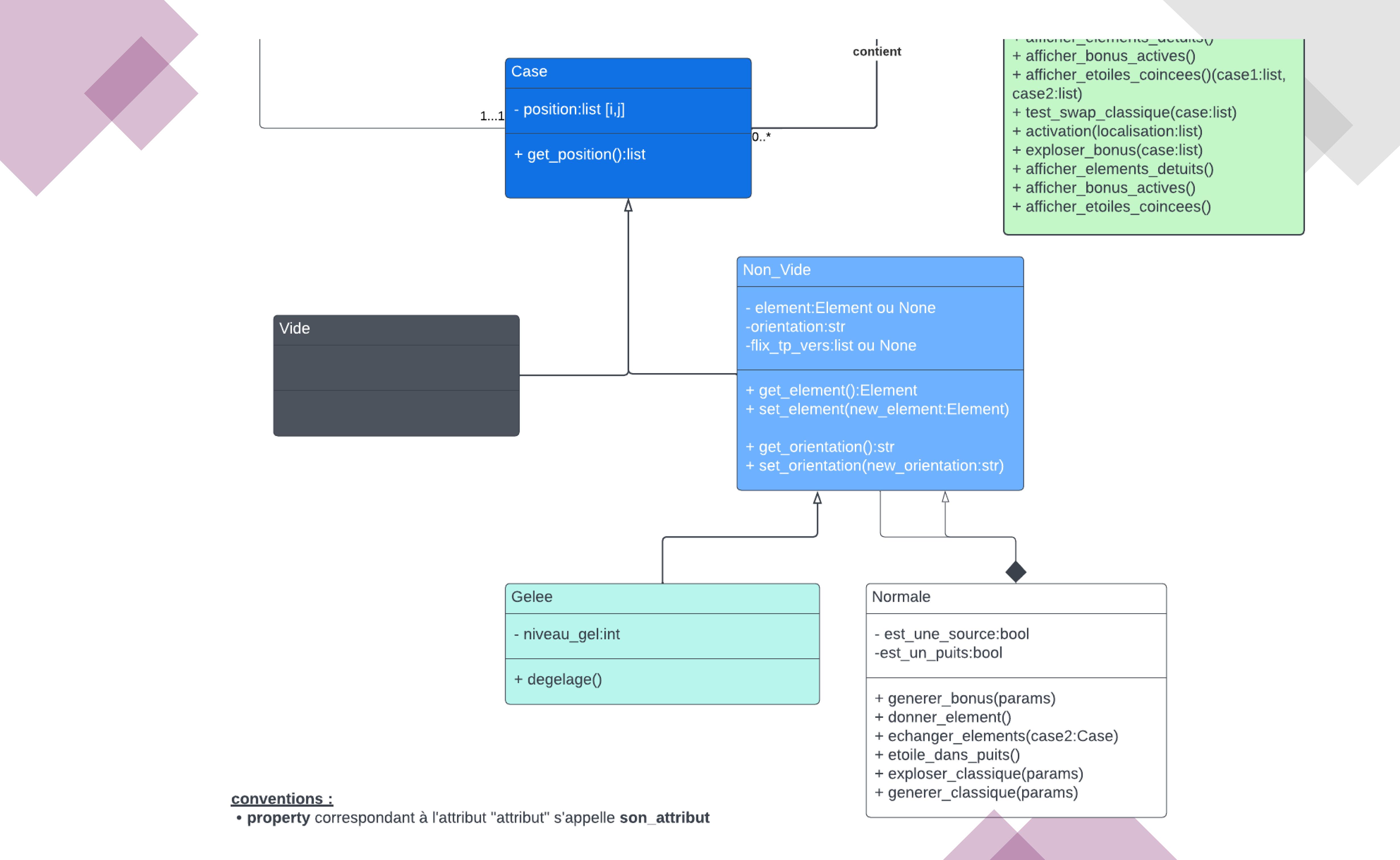
Présentation Projet MiniPOO MINI POO: Match-3

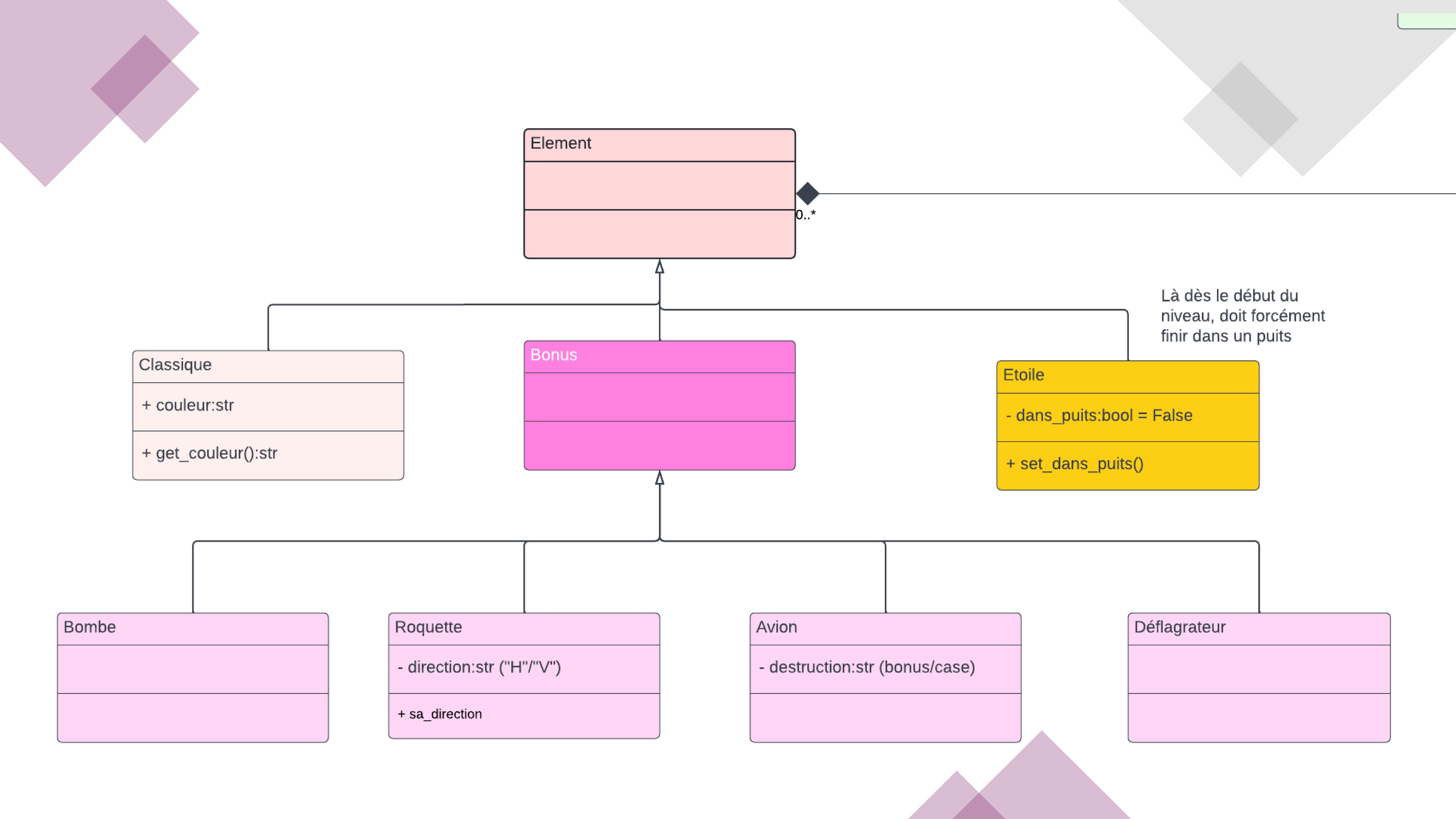
#### Eva GUILBERT, Clotilde GUYARD-GILLES

- Conception des classes et de leurs attributs
- Chargement XML des niveaux
- Identification de combinaisons
- Déplacement des éléments
- Décompte des objectifs
- Interface (visualisation)







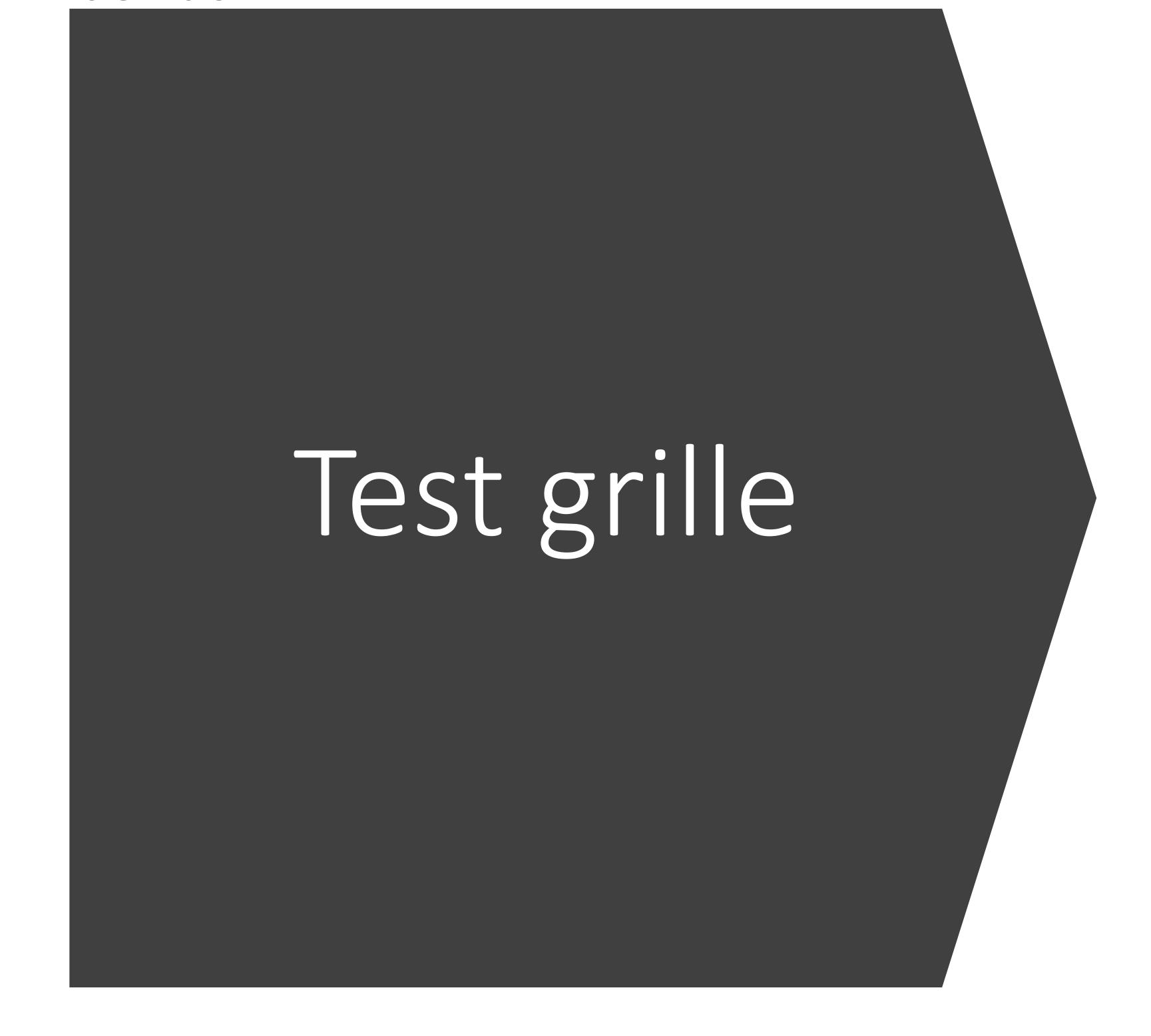


# Déroulé d'une partie classique

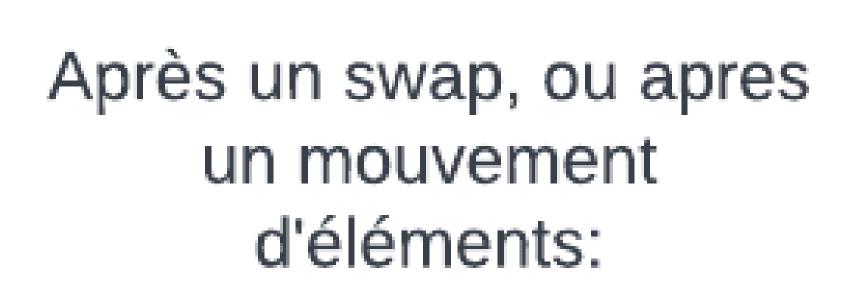
- Chargement de la grille à partir des données XML
- Tester la présence de pattern dans la grille
- =>si oui: supprimer, ajouter des bonus, faire "couler" les éléments
- =>boucle test pattern & mouvement des éléments jusqu'à avoir une grille pleine
- Effectuer un swap
- Tester si un pattern a été formé
- =>si oui: supprimer, ajouter des bonus
- Faire "couler" les éléments, =>boucle test pattern & mouvement des éléments jusqu'à avoir une grille pleine...
- Effectuer un swap...

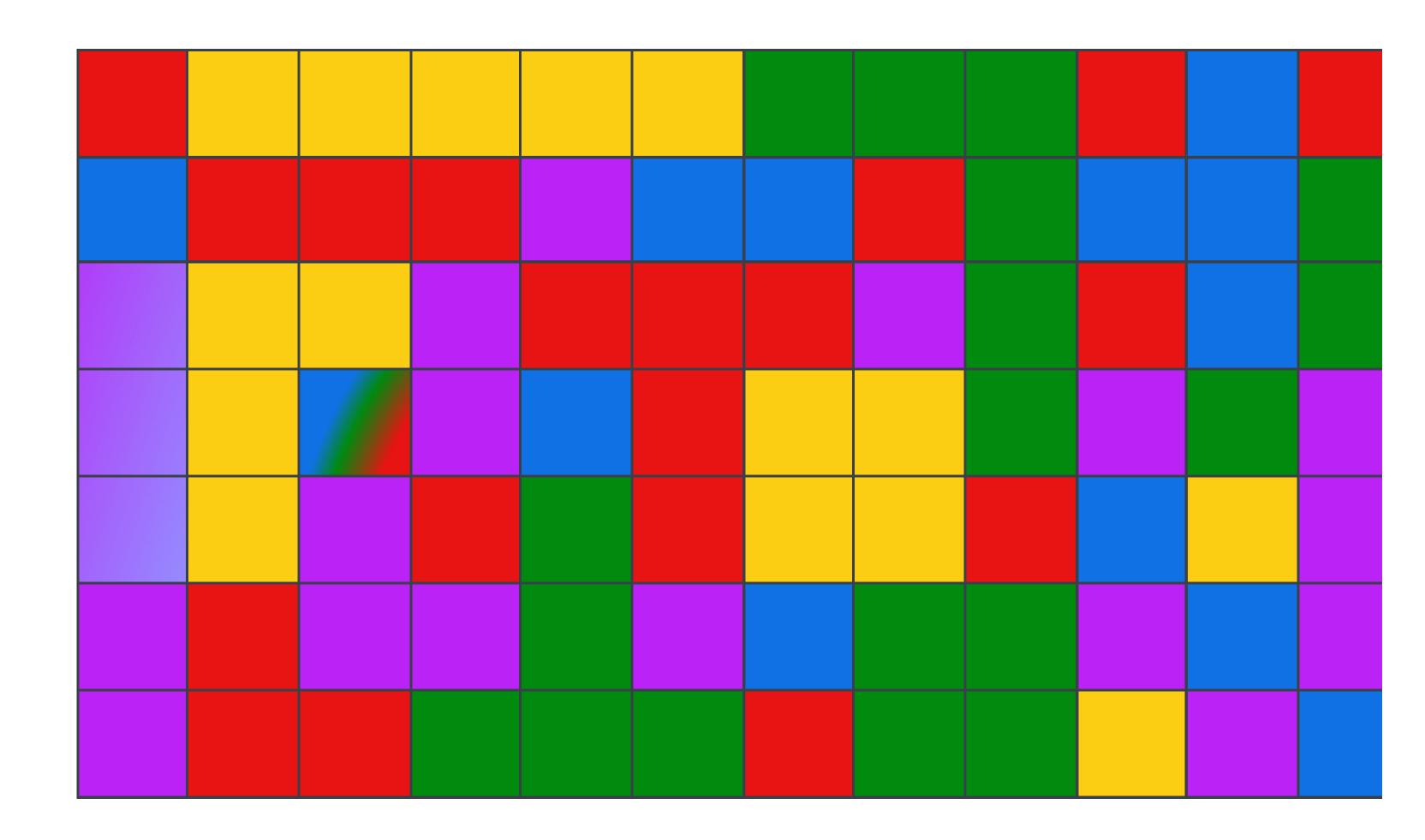
```
-chargement_XML()
-renouveler_grille()
->supprime les
patterns
->remplit et vide le
tableau jusqu'à ce
qu'il soit plein
-test_swap()
->supprime les
pattern formés
-renouveler_grille()
```

Mission: Repérer toutes les nouvelles combinaisons d'une grille nouvellement générée, Respecter la hiérarchie des bonus



-Pour chaque Bonus: parcourt les lignes, puis les colonnes -teste si la case est Normale/Gelée, l'élément Classique -détruit les éléments classiques, dégele les cases Gelées, pose des bonus



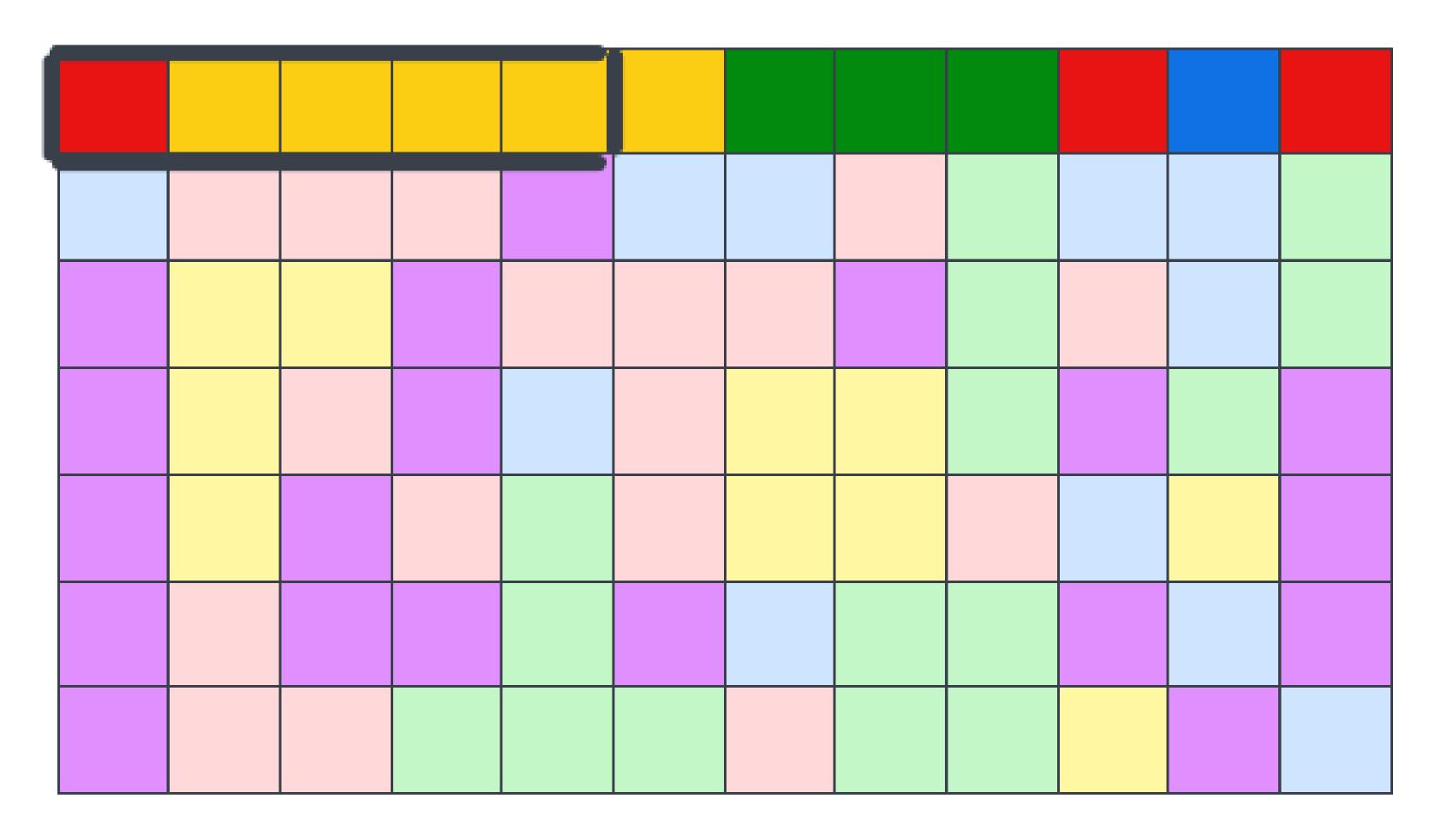


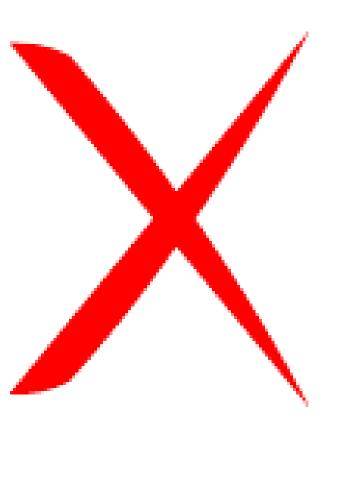
test\_grille()

principe: parcourir ligne par ligne, colonne par colonne la grille, afin de detecter des assemblages

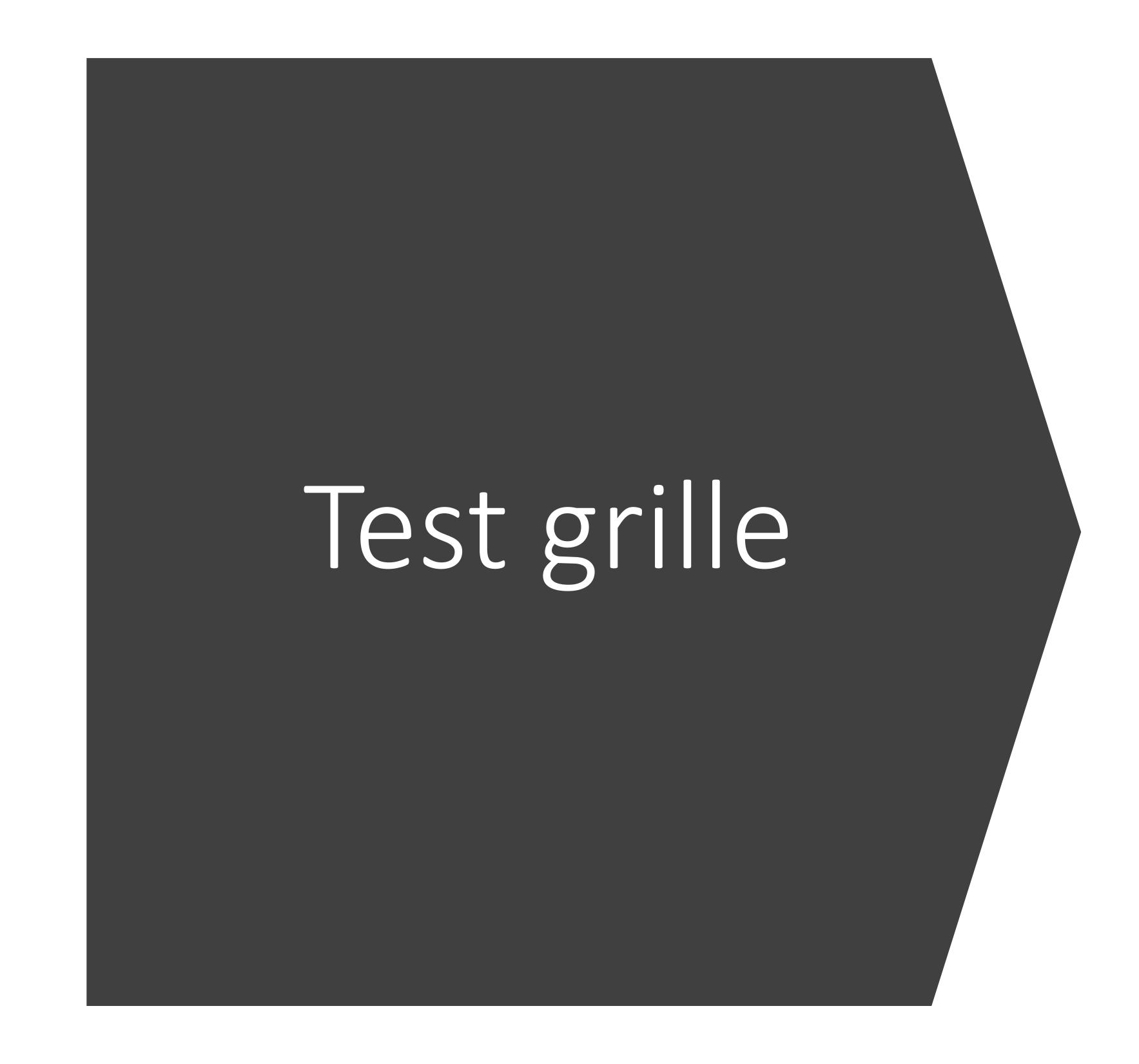
1er parcours:5 cases: Deflagrateur

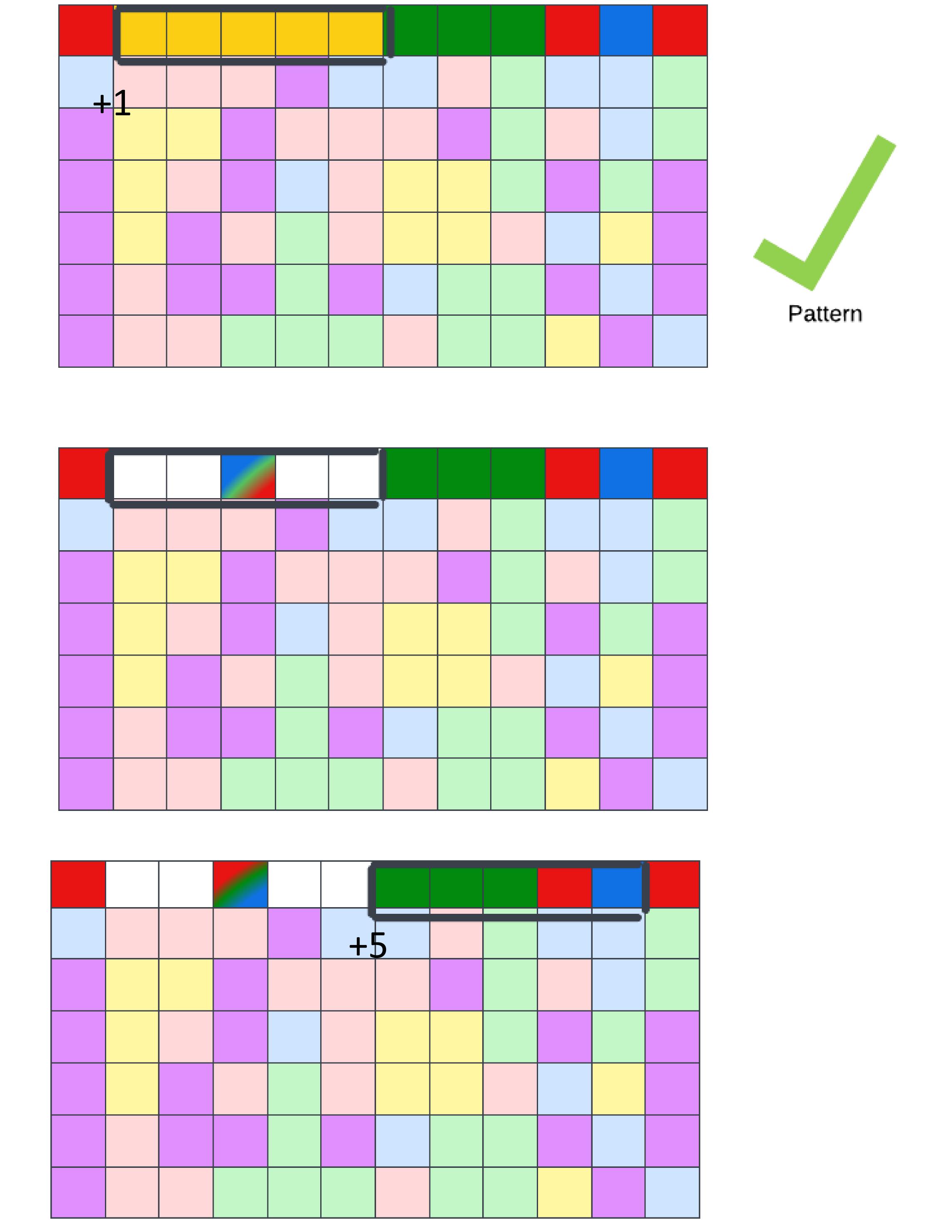
1.a: Horizontalement





pas de pattern

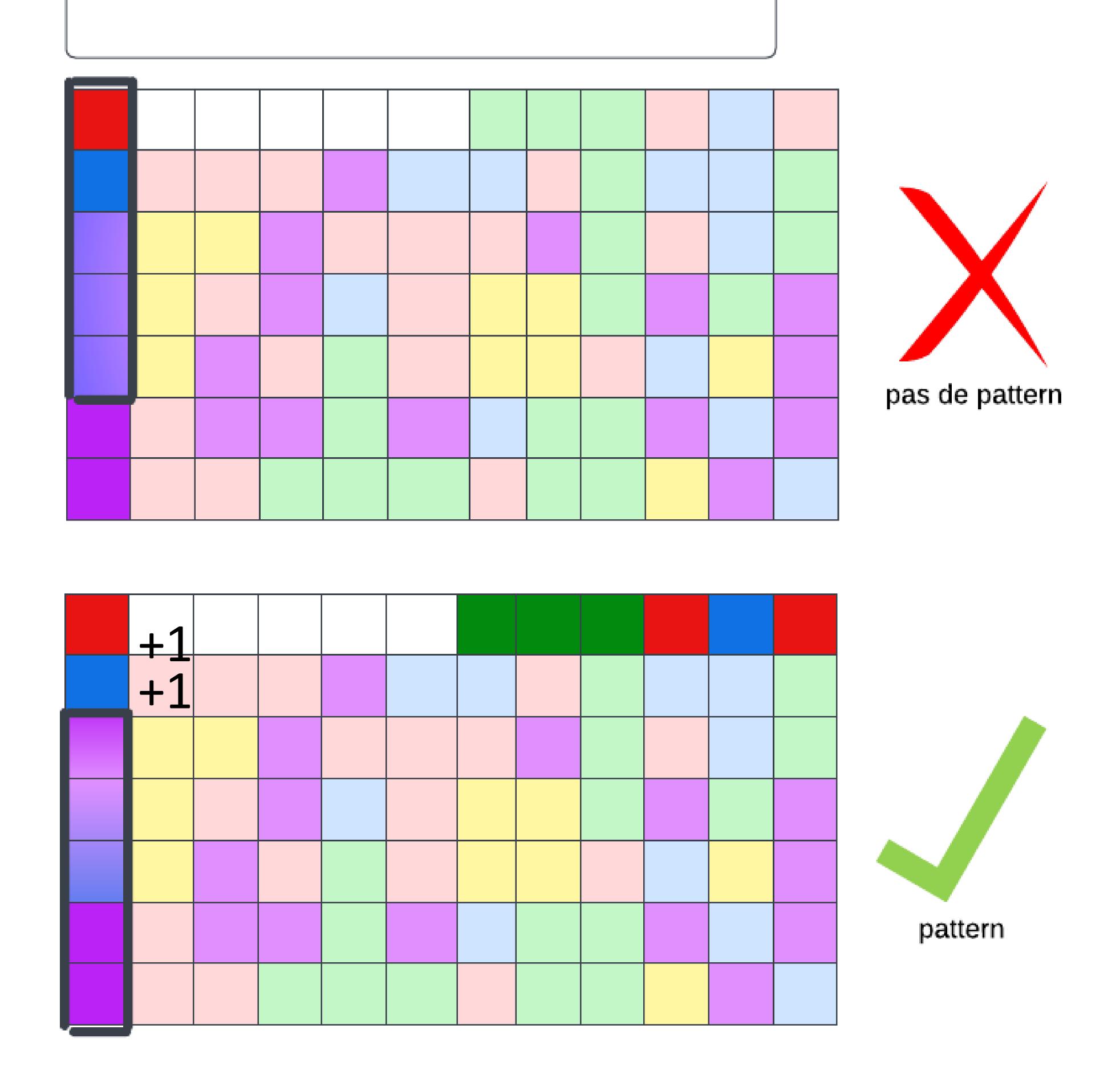


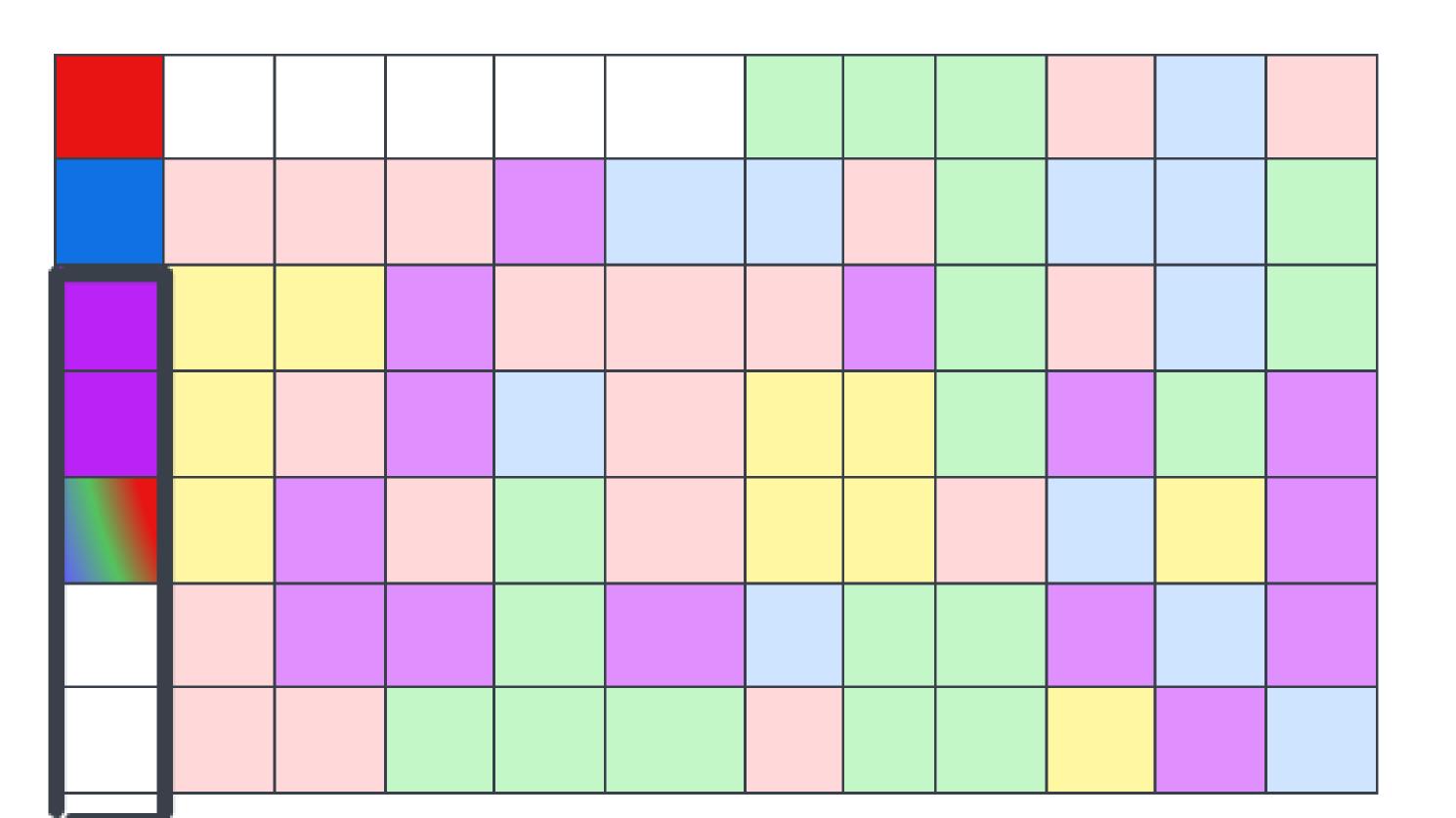


# Test grille

1er parcours:5 cases: Deflagrateur

1.b: Verticalement

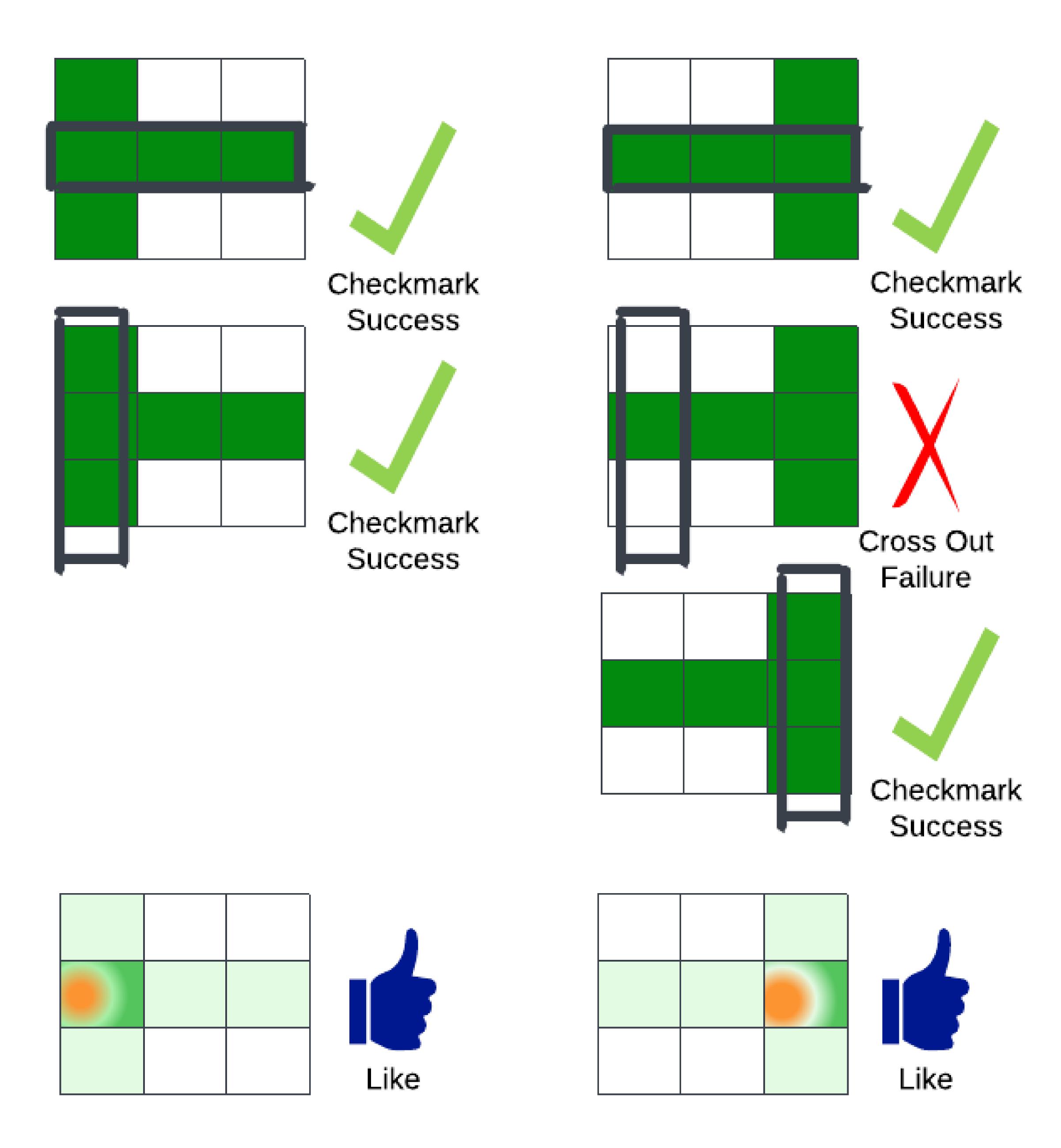


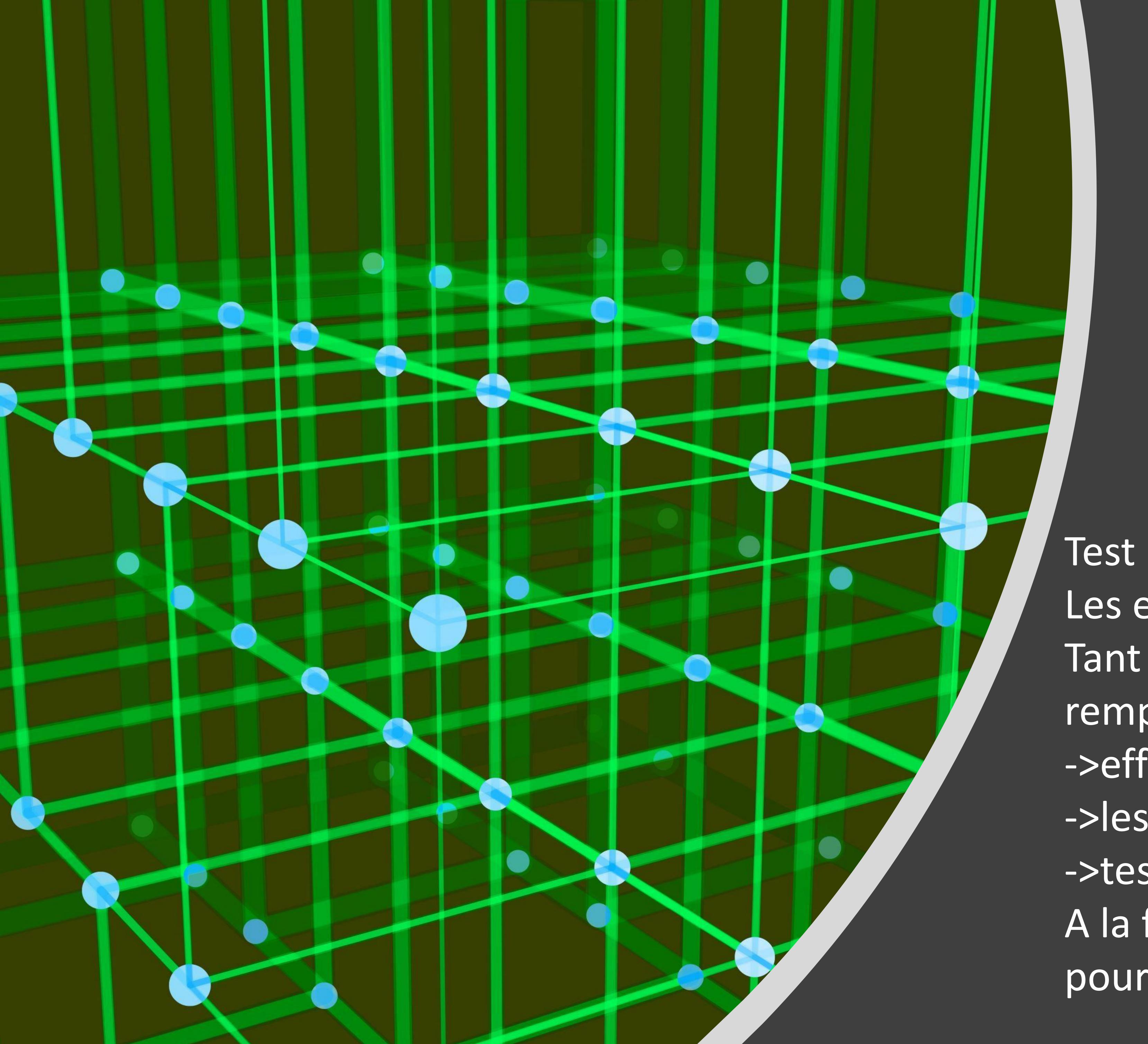


Cas de la Bombe Exemple: bombe 'horizontale'

principe: tester si 3 cases sont alignées, puis tester si 2 cases les entourent au debut ou à la fin

Test grille





# Renouveler\_grille

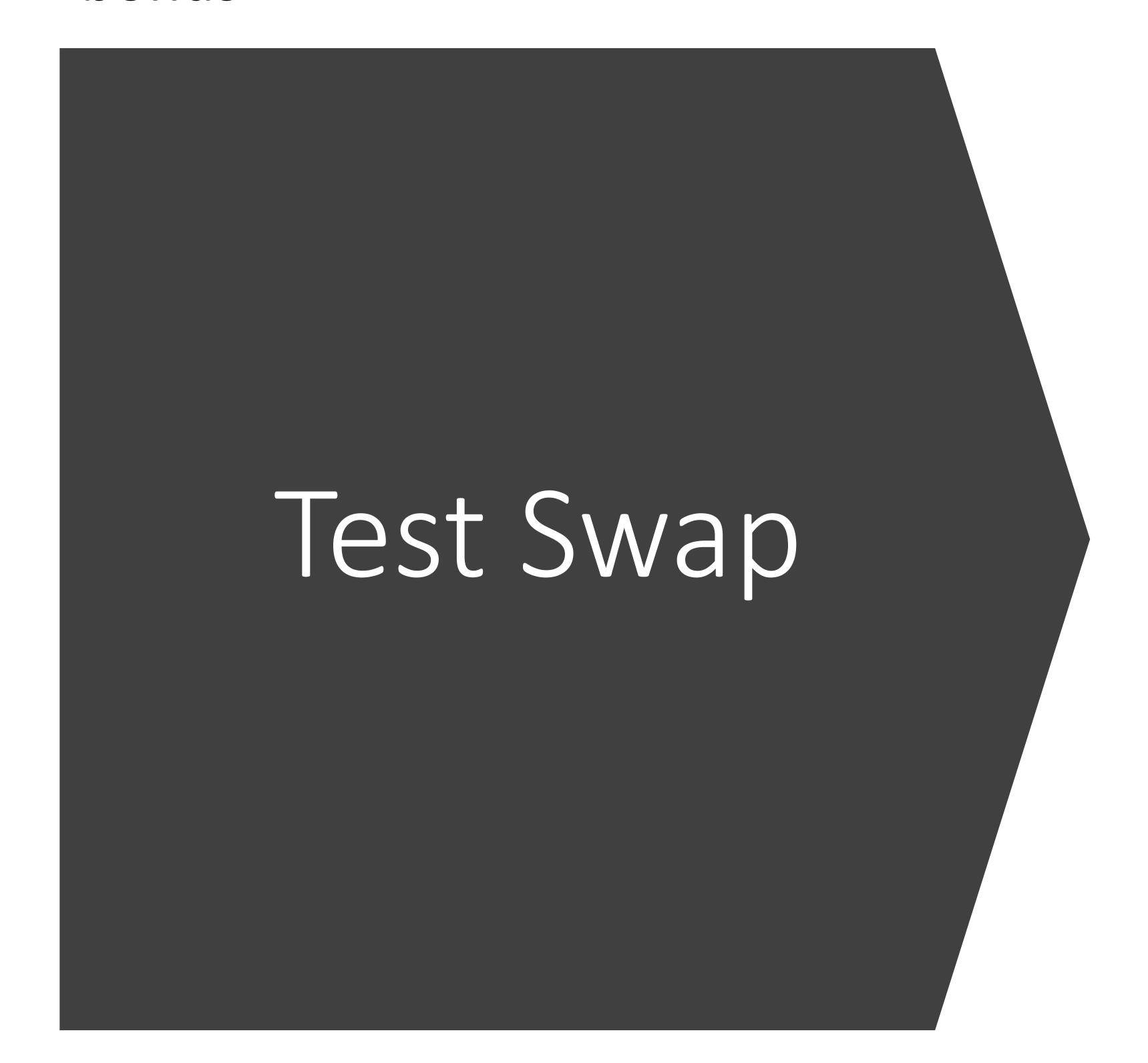
Test les patterns dans la grille (test\_grille()) Les enlève, pose des bonus...

Tant que la grille n'est pas entièrement remplie:

- ->effectue le flux des éléments
- ->les cases source génèrent des éléments
- ->test les patterns...

A la fin, la grille est prête à être utilisée pour un swap

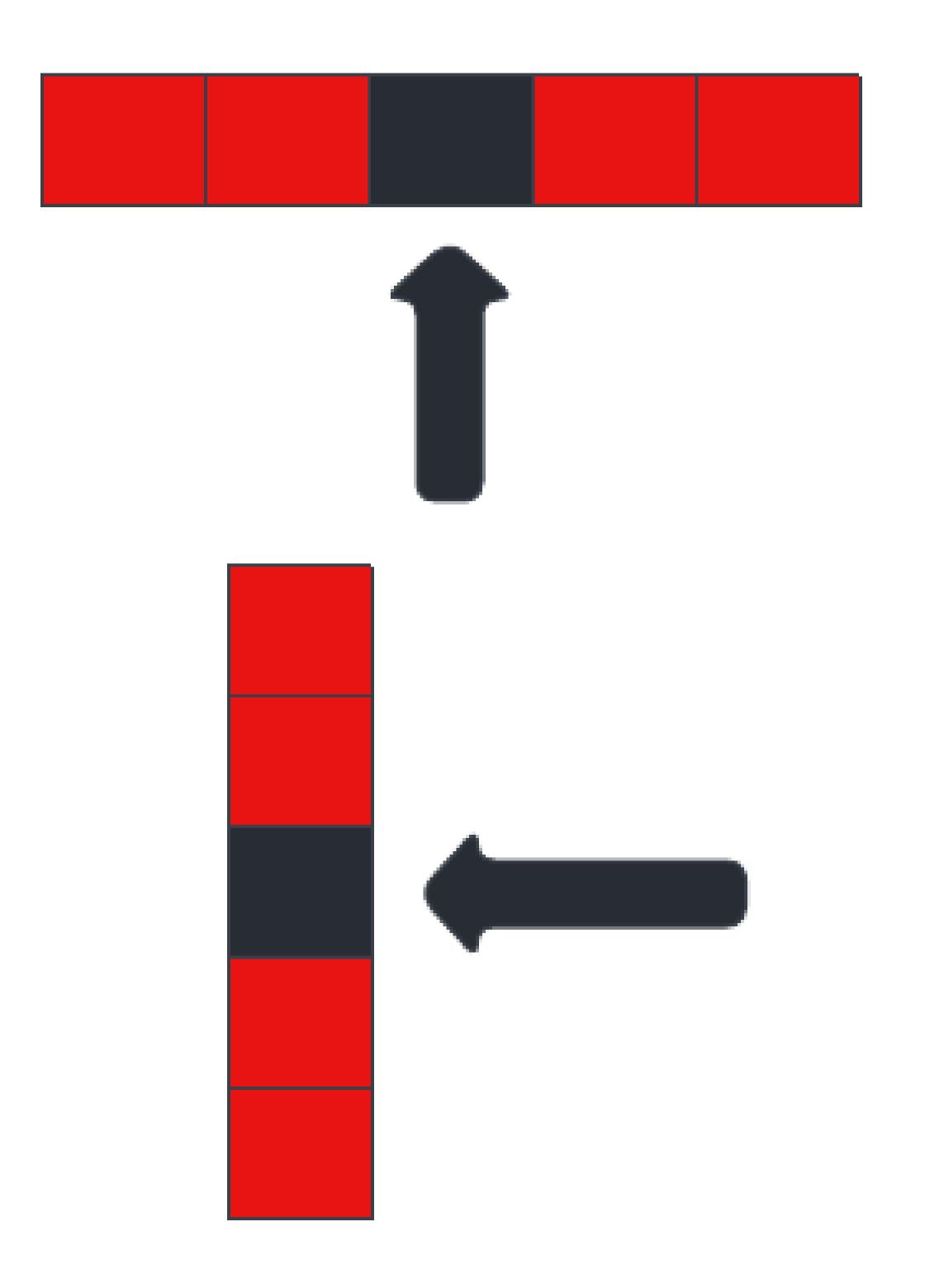
Mission: tester toutes les combinaisons possibles en respectant la hiérarchie des bonus



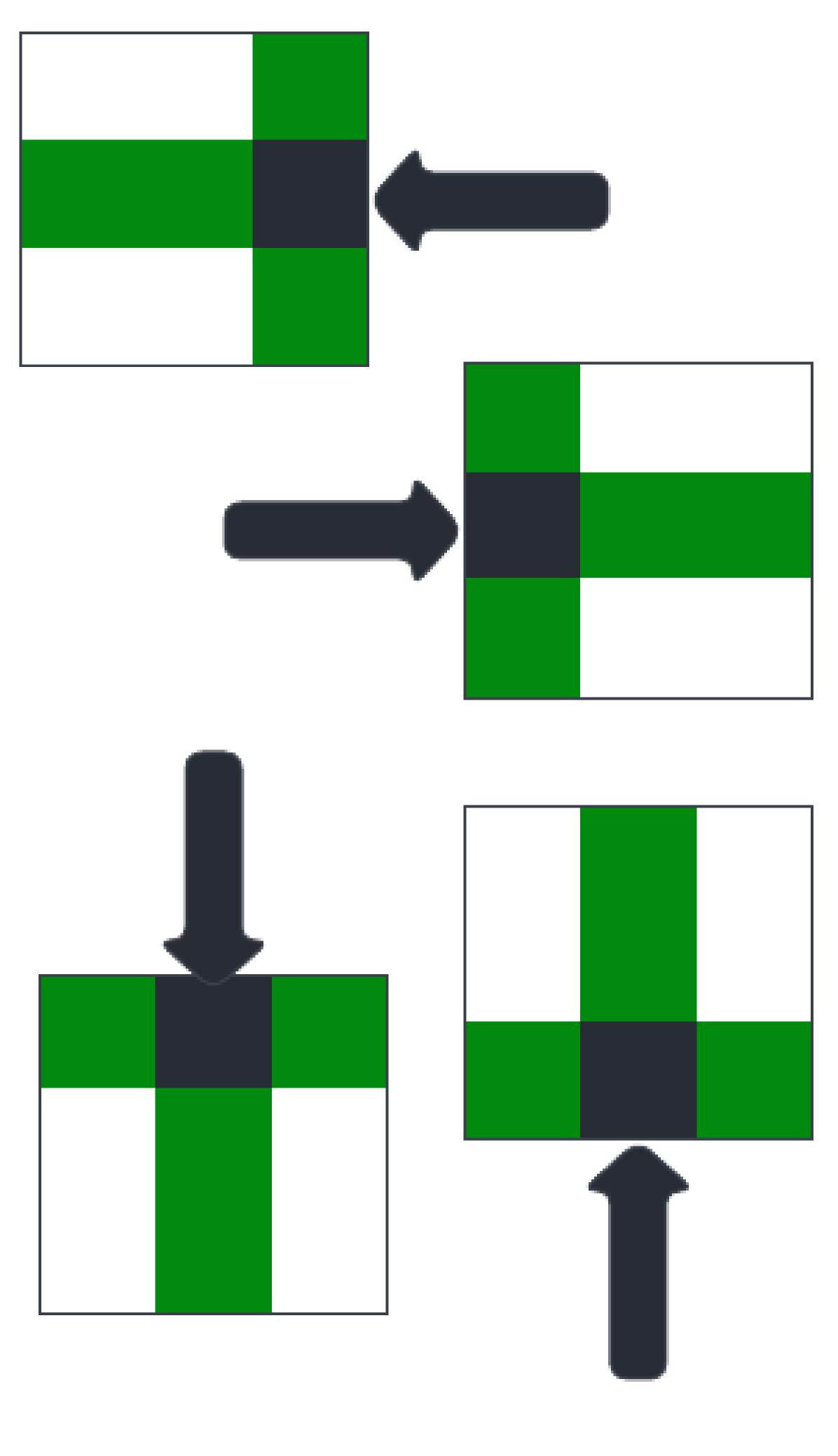
- -Vérification cases Non Vides, éléments classiques, de même couleur
- -Destruction éléments Classiques, dégelage cases Gelées, ajout d'un bonus

Cas d'un swap:

déflagrateur:

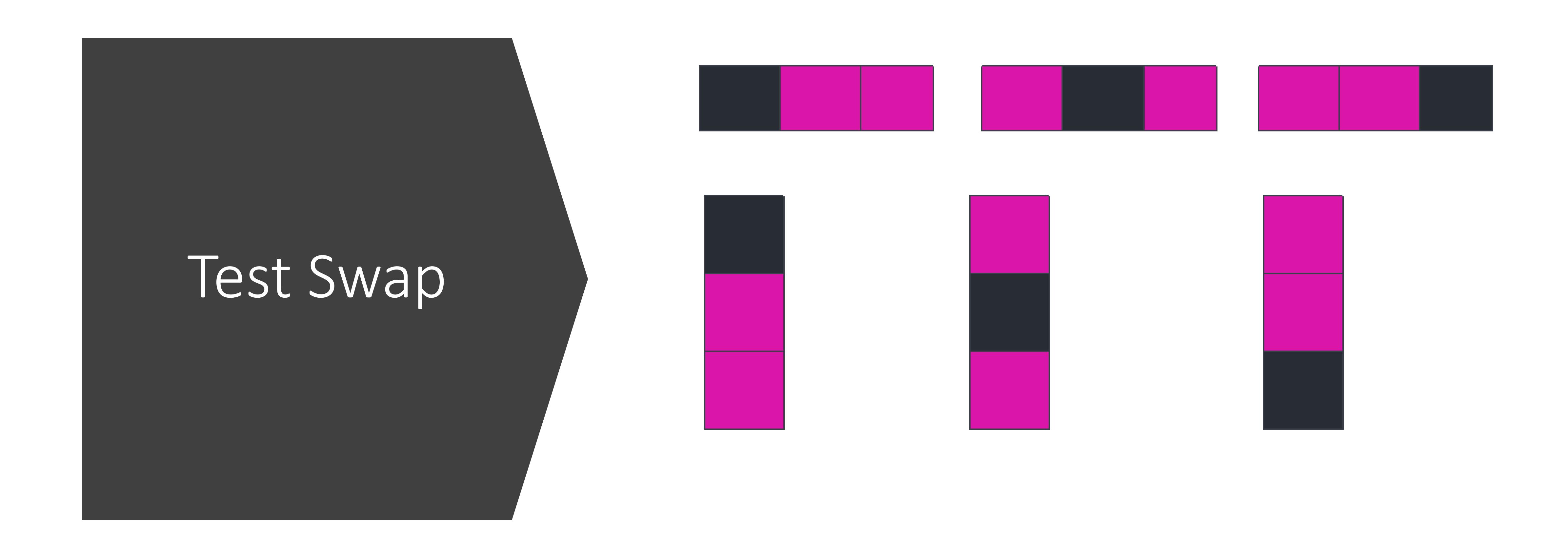


Bombe:



Roquette: Avion: Test Swap

Match 3:



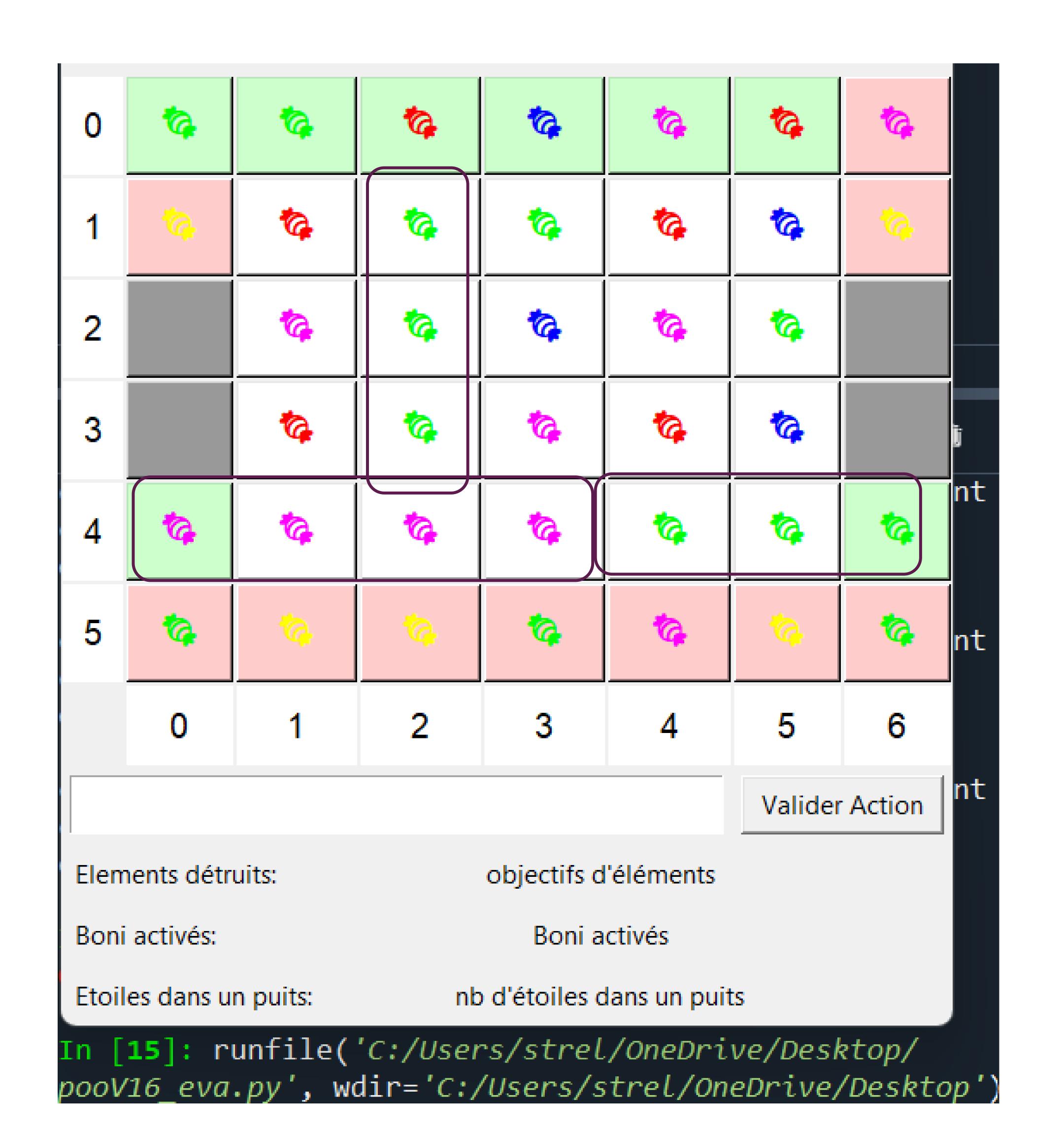
## INTERFACE

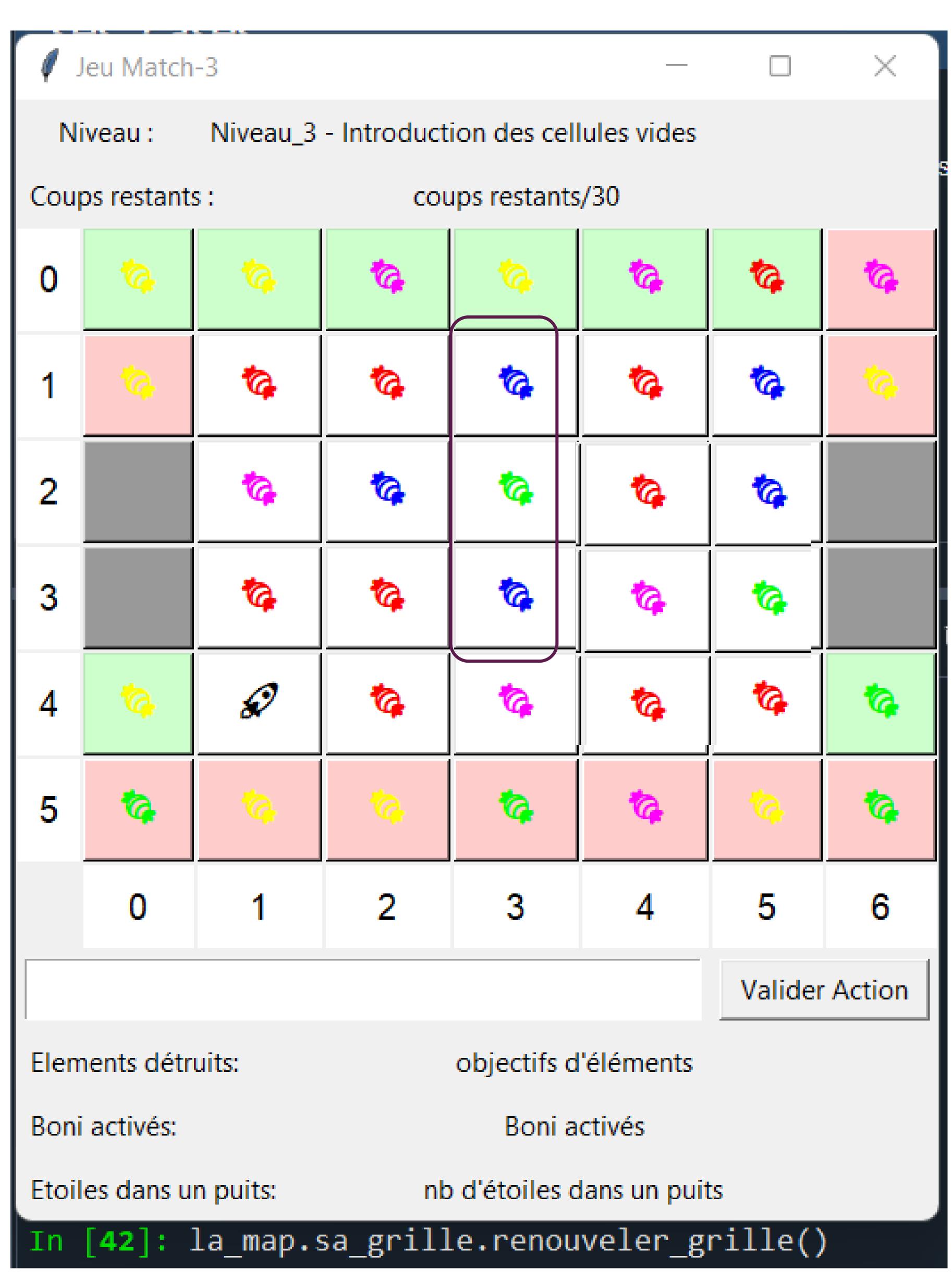


## INTERFACE:

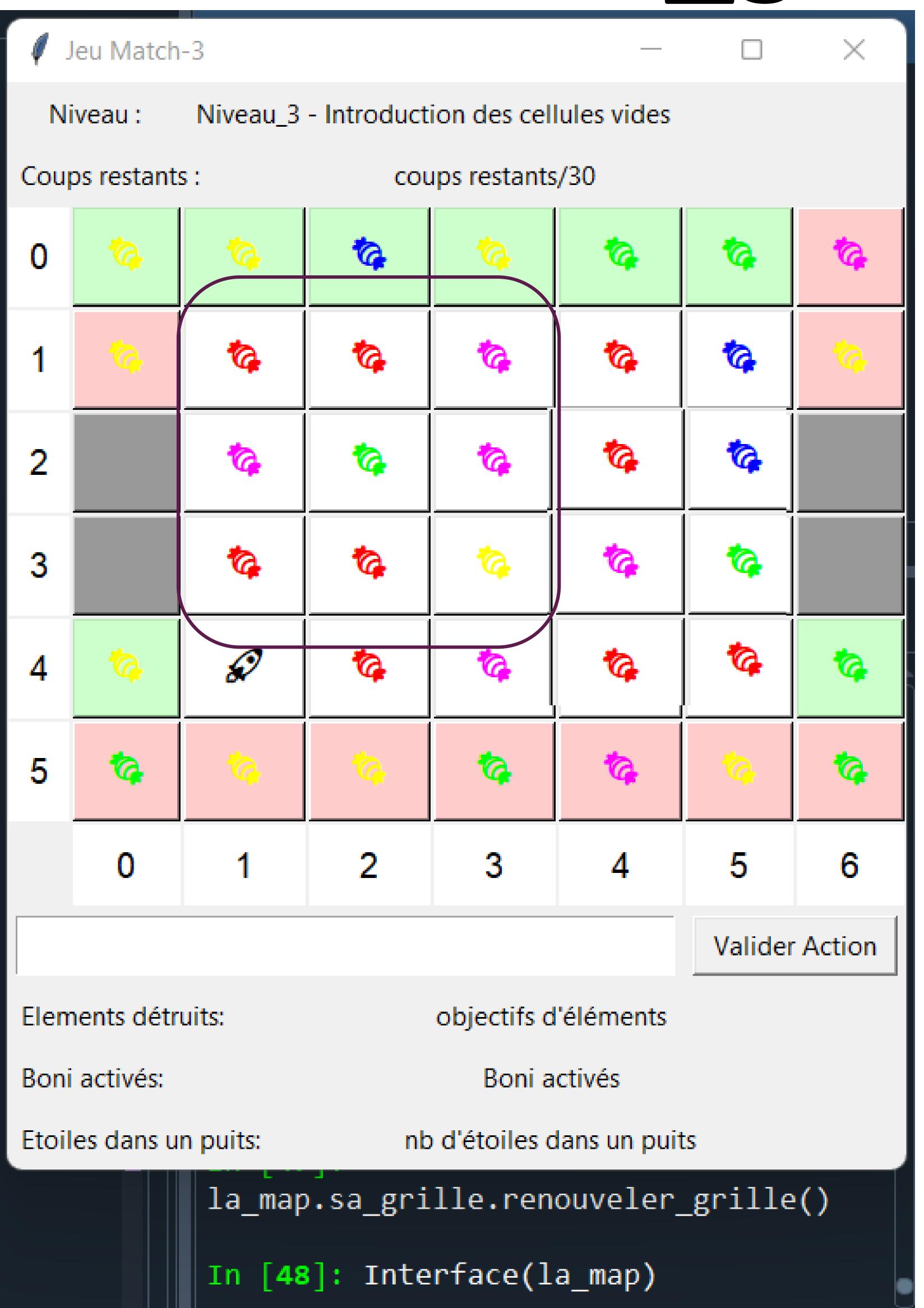


# Test de la fonction test\_swap() & renouveler grille()

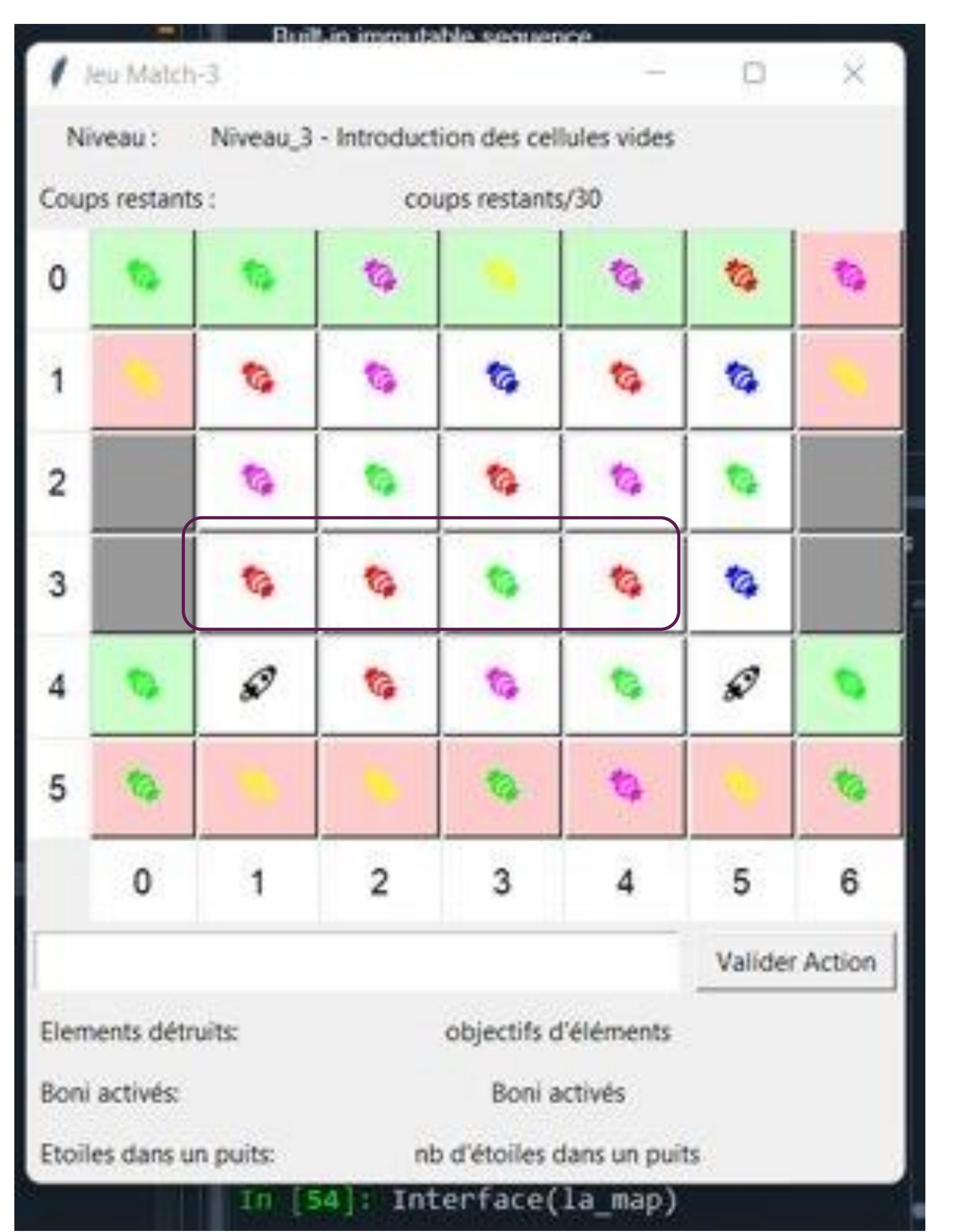


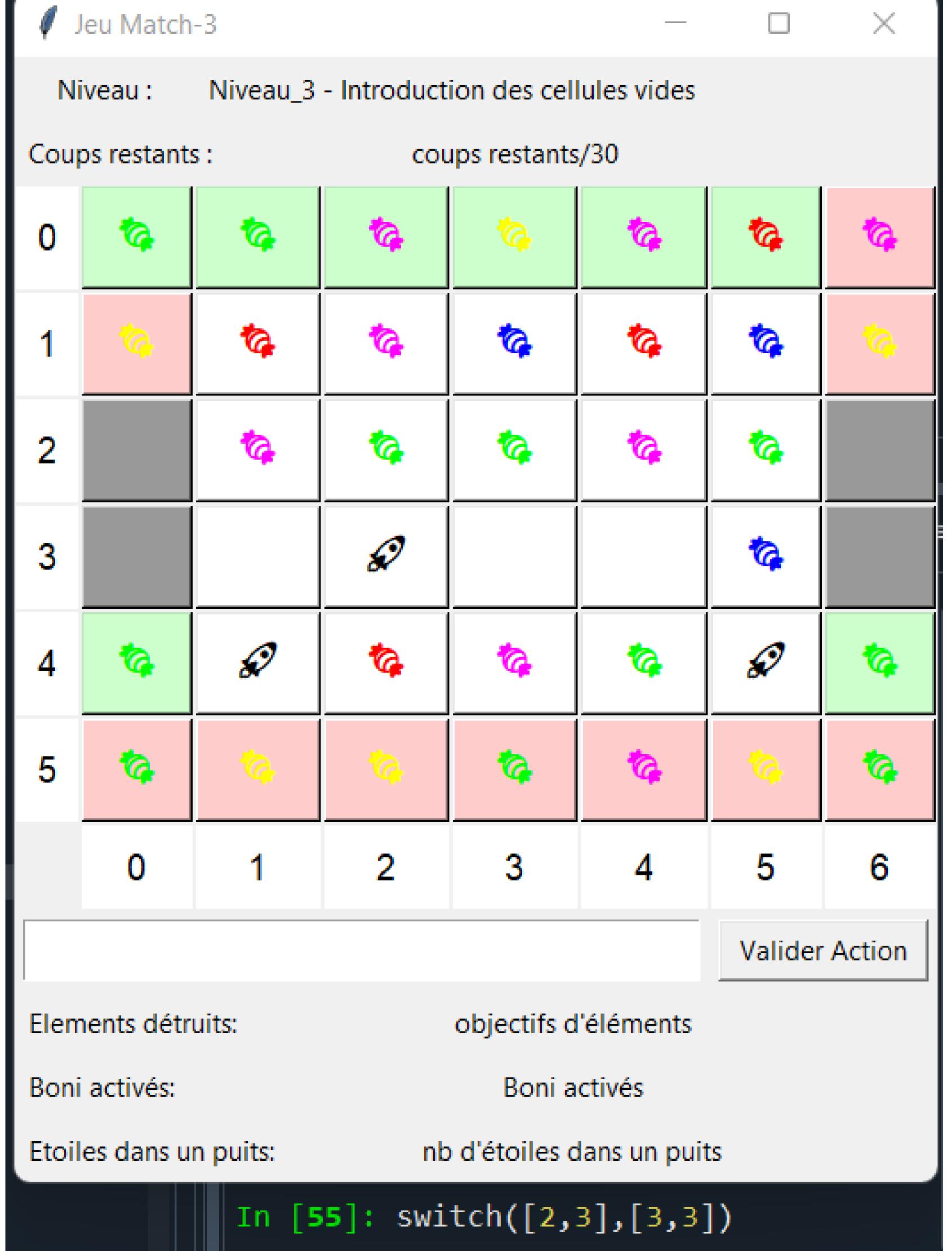


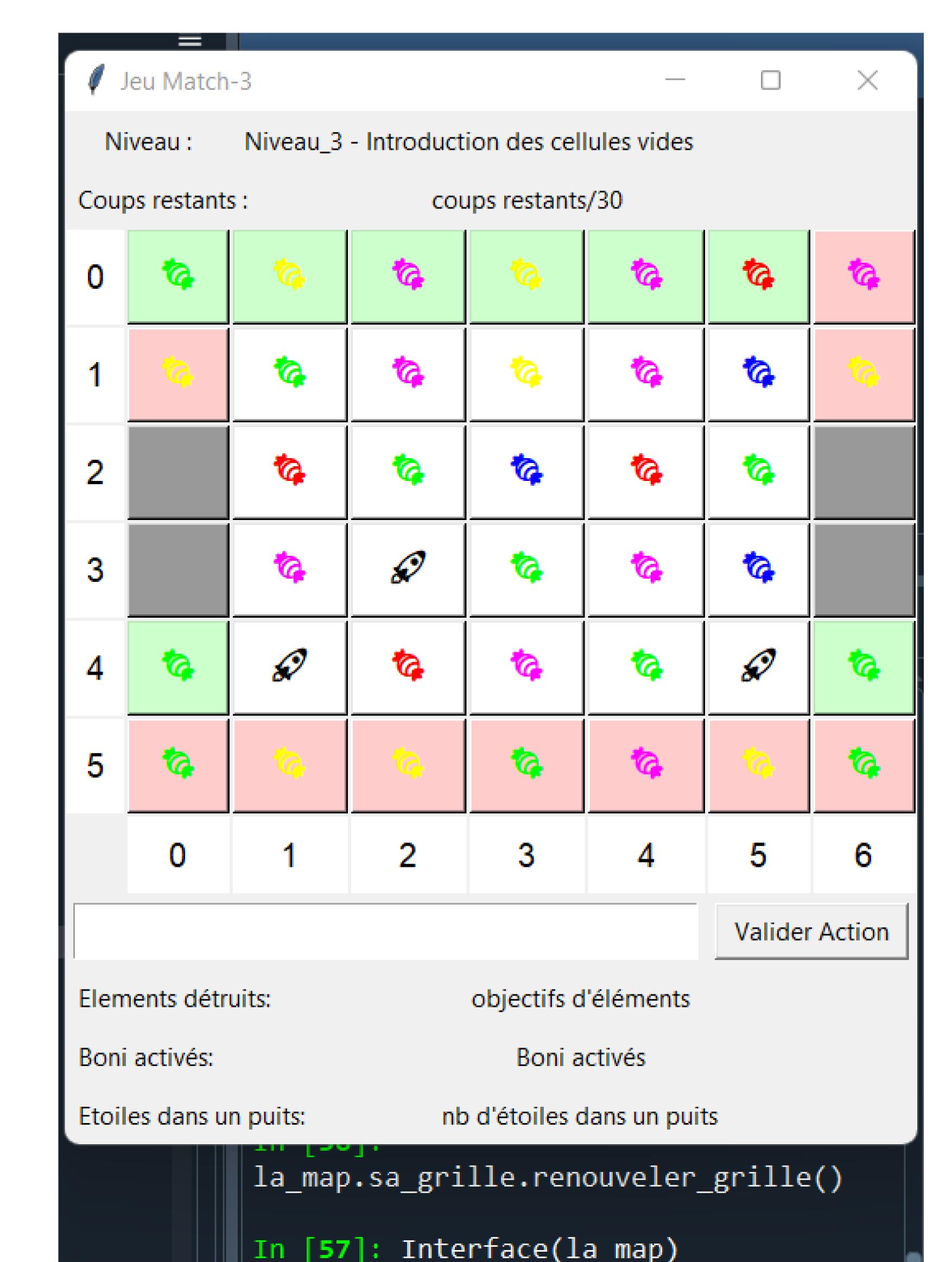
Test de la fonction test\_swap() & renouveler\_grille()



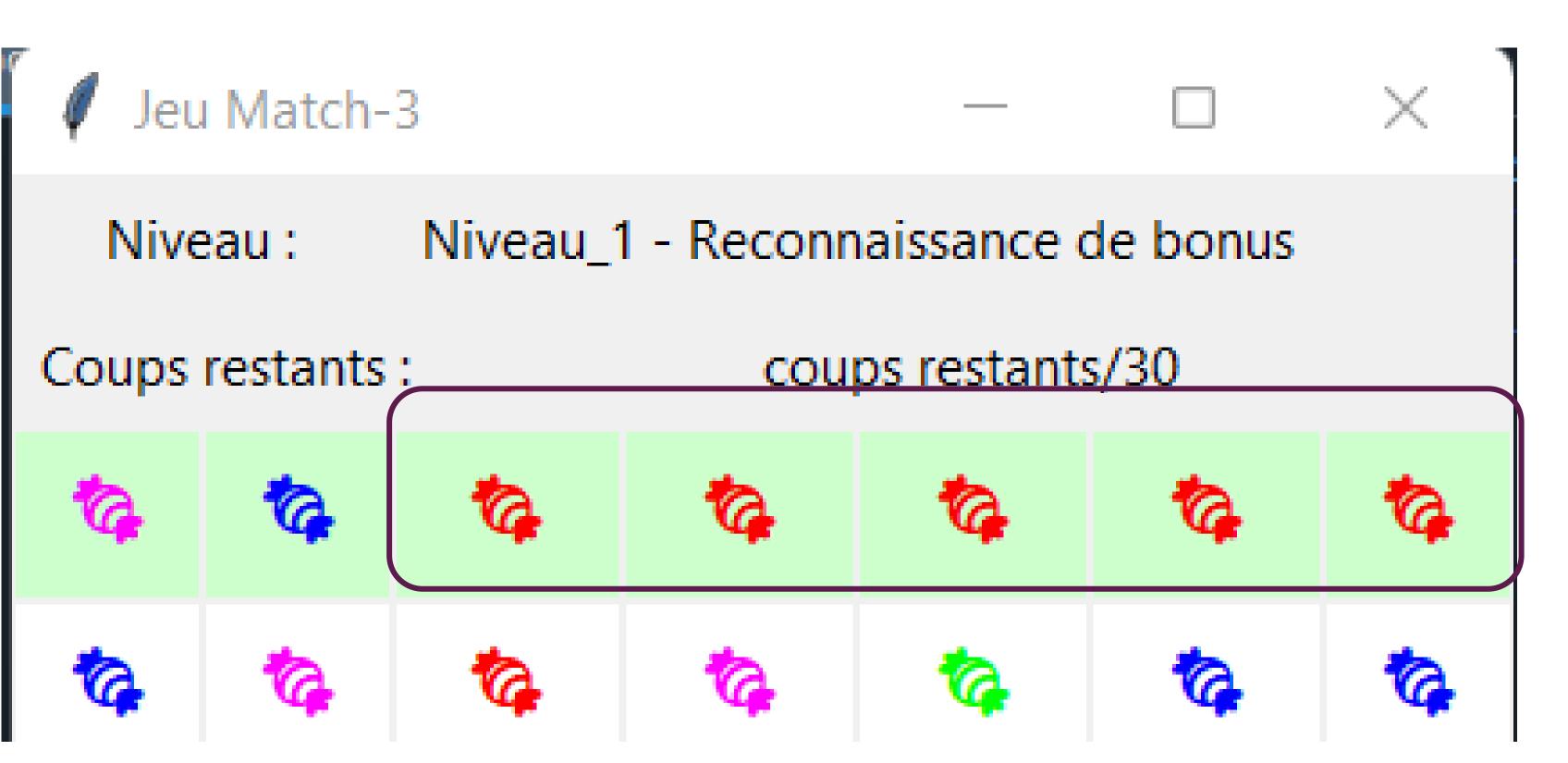
# Tester switch (case1, case2)





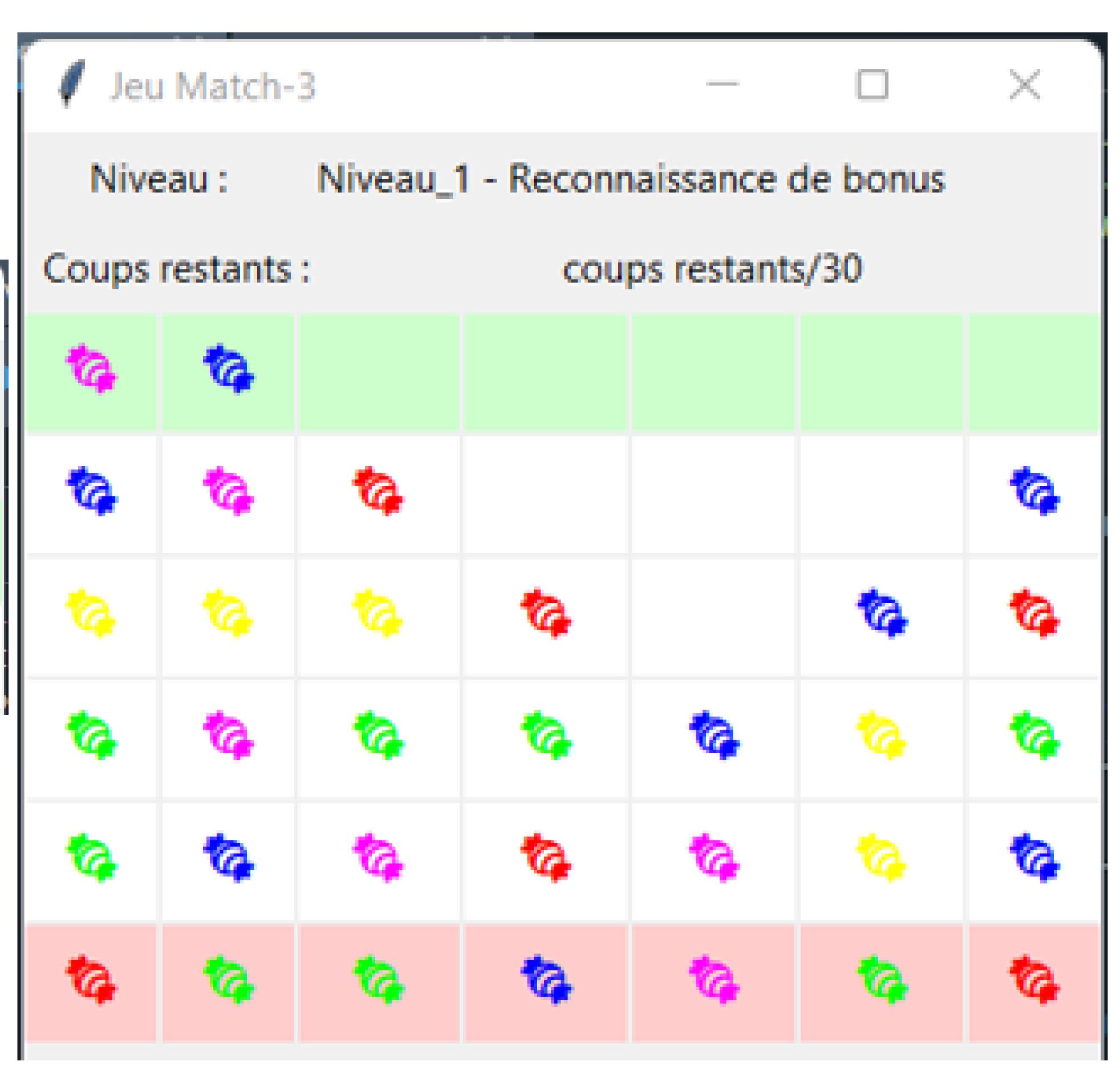


# Création et explosion d'un bonus





la\_map.sa\_grille.test\_swap(case1, case2)



Exploser\_bonus([0,4])

# Statistiques:

Environ 1300 lignes, avec Chargement XML, définitions des classes et interface

#### Classe Map

- 7 fonctions
- Nombre de coups, couleurs, définition/ajout objectifs, dimensions, titre...
- Environ 40 lignes

#### Classe Objectifs

- 3 fonctions
- Définition cible, quantité, REPR

# Statistiques

#### Grille

- 14 fonctions
- Instanciations & \_get \_set
- Activer des cases, exploser des bonus, tester si un swap est valide, tester des cases de même couleur, tester les assemblages dans la grille, renouveler la grille...
- Environs 650 lignes

De très longues fonctions pour tester les différents pattern, en respectant la hiérarchie des Bonus.

Des fonctions plus courtes qui combinent ces longues fonctions

# Statistiques

#### Classe Case

- Classe mere
- 2 fonctions
- Get\_position etREPR

#### Classe Vide

- Est simplement vide
- 1 fonction: REPR

#### Classe Non Vide

- Rassemble les éléments communs aux cases Normales et Gelées
- 3 fonctions:
   orientation, téléportation, \_get & \_set élément

# Statistiques

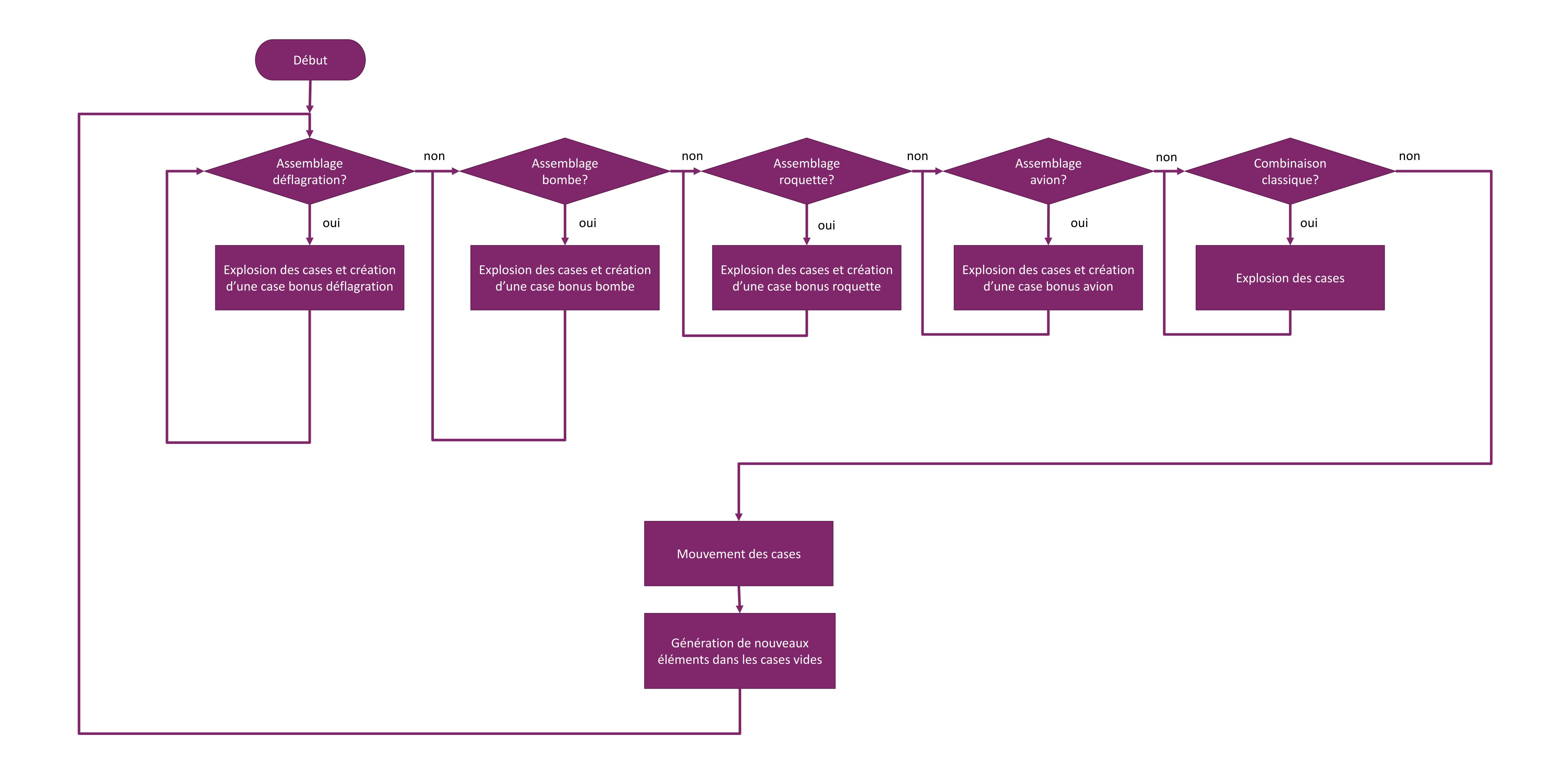
#### Classe Normale

- 8fonctions
- Générer et exploser un élément classique, donner un élément selon le flux, échanger ses éléments
- Environ 70 lignes

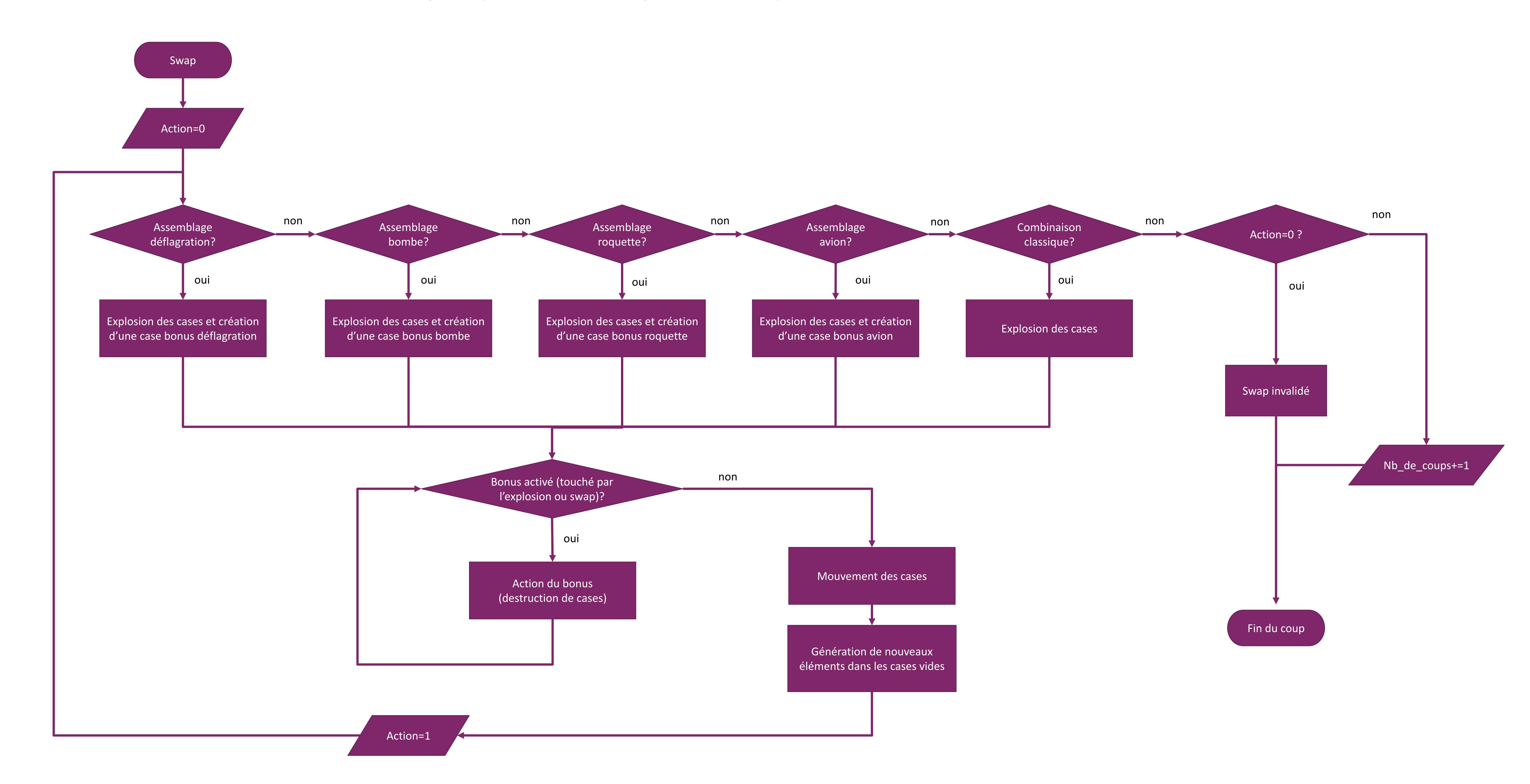
#### Classe Gelée

- 3 fonctions
- \_get & \_set niveau gel, diminuer un niveau ou dégeler completement, REPR

#### Traitement après mouvement et génération de nouveaux éléments



#### Traitement des assemblages après action du joueur (swap)



#### Traitement des assemblages après action du joueur (clic d'activation d'un bonus)

