



Arthur Correnson (CodeAnon 2020)

MINI-CONFÉRENCE/ATELIER

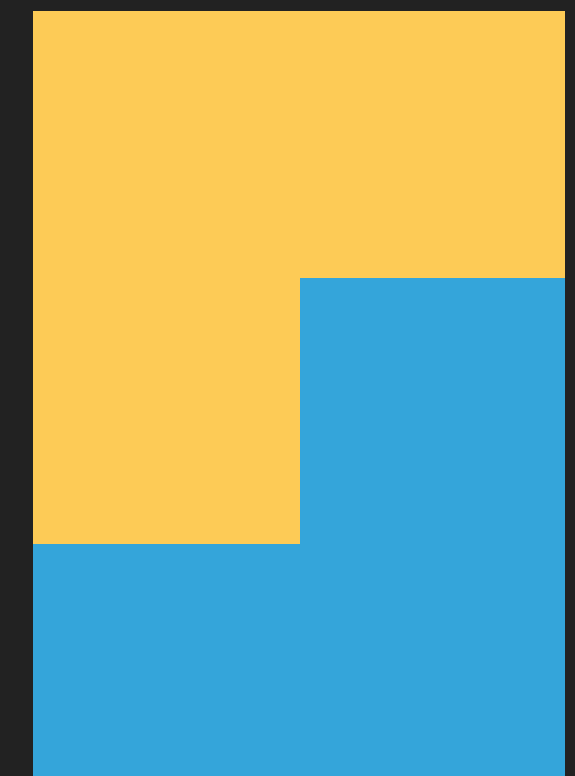
DIVISER POUR RÉGNER, TETRIS, PROGRAMMATION RÉCURSIVE

LE PROBLÈME

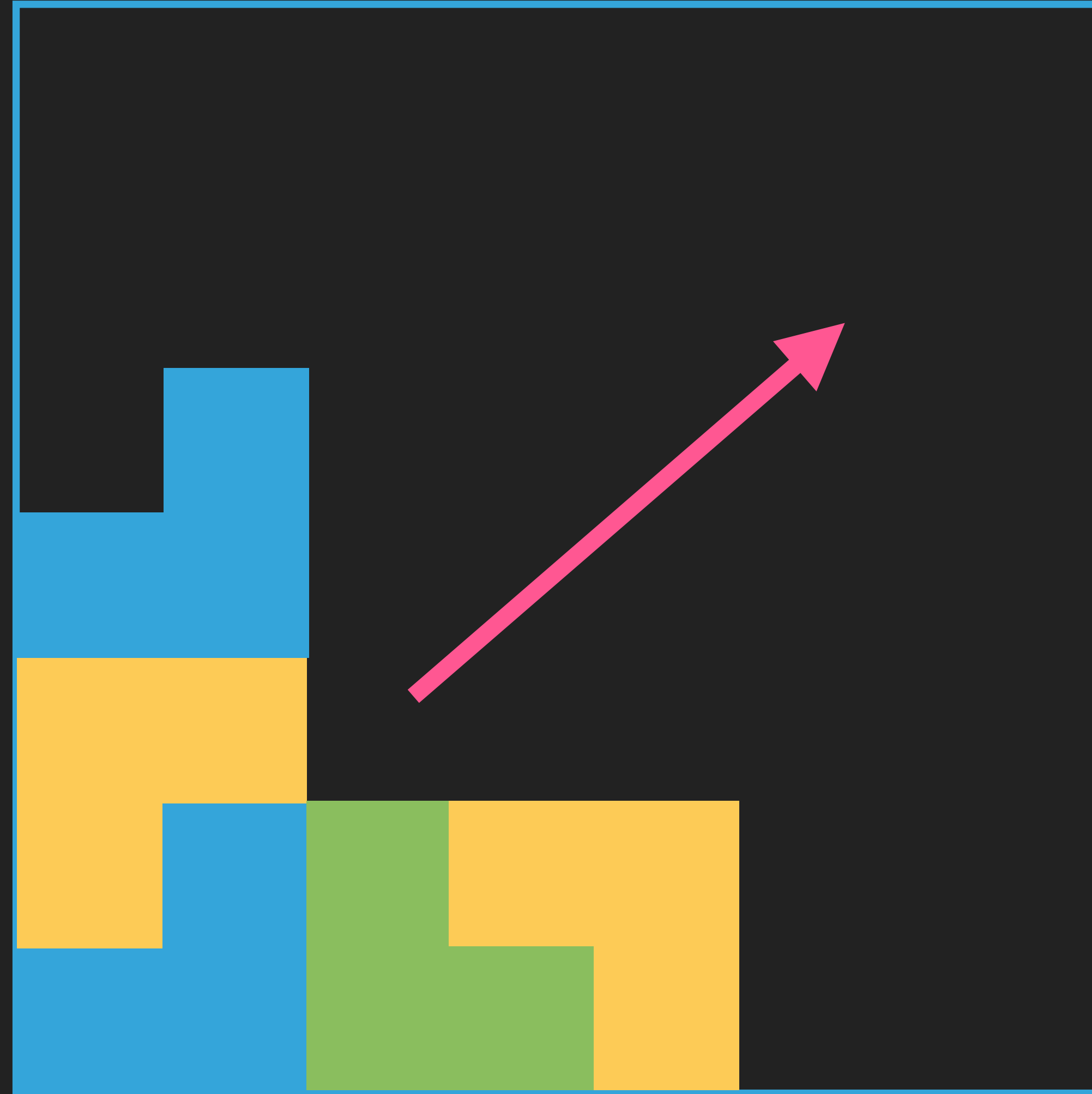
- ▶ On se donne un carré de côté 2^k
- ▶ On se donne une « pièce de puzzle »
- ▶ On cherche à paver le carré :
 - ▶ avec uniquement cette pièce de puzzle
 - ▶ en laissant 1 case vide (la case est imposée)
 - ▶ en un temps raisonnable



Une pièce



Deux pièces imbriquées



STRUCTURE DU PROBLEME

- ▶ On cherche à programmer une fonction `PUZZLE(n, x, y)`
 - ▶ Qui résout le problème sur un carré de taille 2^n
 - ▶ La case laissée vide est la case x, y
 - ▶ La fonction retourne une matrice carré de taille 2^n représentant le carré pavé

COMMENT FAIRE ?

LES DIFFICULTÉS

- ▶ Comment résoudre le problème d'un point de vue théorique
- ▶ Comment résoudre le problème d'un point de vue pratique
 - ▶ Comment représenter le problème en mémoire
 - ▶ Ecrire une procédure de résolution efficace
 - ▶ Ecrire un code lisible et facilement compréhensible

**(UNE) SOLUTION EN
OCAML**

POURQUOI EN OCAML ?

- ▶ Langage original
 - ▶ Peu de gens programment en OCaml mais il est quand même utilisé dans l'industrie (Facebook...)
- ▶ Langage *safe*
 - ▶ Typage fort (évite les erreurs de programmation)
 - ▶ Pas de problèmes de mémoire (accès indésirables dans des tableaux...)
- ▶ Langage élégant
 - ▶ Donne du code très propre et très lisible
 - ▶ Proche du raisonnement mathématique (utile pour découper des problèmes un peu théorique comme le notre)

LET'S CODE !