



# *Challenge Moteur de jeux*

**IG2I**  
INSTITUT DE GENIE INFORMATIQUE ET INDUSTRIEL

**LE1 2017-2018**

Challenge Moteur de jeux - 2018

1 CENTRALE LILLE



# Plan

- Introduction : Objectifs
- Règles du jeu d'Avalam
- Ressources
- Livrables
- Soutenance
- Perspectives



# *Introduction*

## Programme & Objectifs



# Bruits de couloir...

- “C’était mieux avant, ils ne savent plus réfléchir”
- “Il faut des pédagogies actives”
- “Pour eux, l’informatique, c’est les jeux vidéo”
- “Les TNE, ça ne marche pas en première année”

VOYONS ÇA !



# Objectifs de ce module

- Formalisation & Modélisation
  - Définition d'une terminologie
  - Passage de l'exemple à la règle
- Algorithmique
- Programmation en C
- Travail par projet, en équipe
- + Valorisation possible pour la communication de l'ig2i



# Intérêt pédagogique

- Aspect ludique facteur de motivation
- Mode projet susceptible d'offrir à chacun la possibilité d'aller à son rythme
- Concours favorisant l'émulation
- Confrontation d'approches d'enseignants de profils différents



# Intervenants

- Amandine Leriche
  - [amandine.leriche@centralelille.fr](mailto:amandine.leriche@centralelille.fr)
- Nicolas Danloup
  - [nicolas.danloup@univ-lille1.fr](mailto:nicolas.danloup@univ-lille1.fr)
- Nicolas Oxoby
  - [nicolas.oxoby@centralelille.fr](mailto:nicolas.oxoby@centralelille.fr)
- Thomas Bourdeaud'huy
  - [thomas.bourdeaud\\_huy@centralelille.fr](mailto:thomas.bourdeaud_huy@centralelille.fr)



# Intervenants : pourquoi eux ?

- Développer, c'est réfléchir...
- Comment ?
  - Du papier et des crayons ... même en 2018 !
  - Un vocabulaire précis, employé rigoureusement
  - Une démarche : hypothèses / confrontation au réel / généralisation / démonstration
- Tout cela n'est pas le propre des informaticiens...
  - Bien au contraire !



# *Règles du jeu d'AVALAM*

<https://youtu.be/DbZIvQSyFvA>



# Histoire

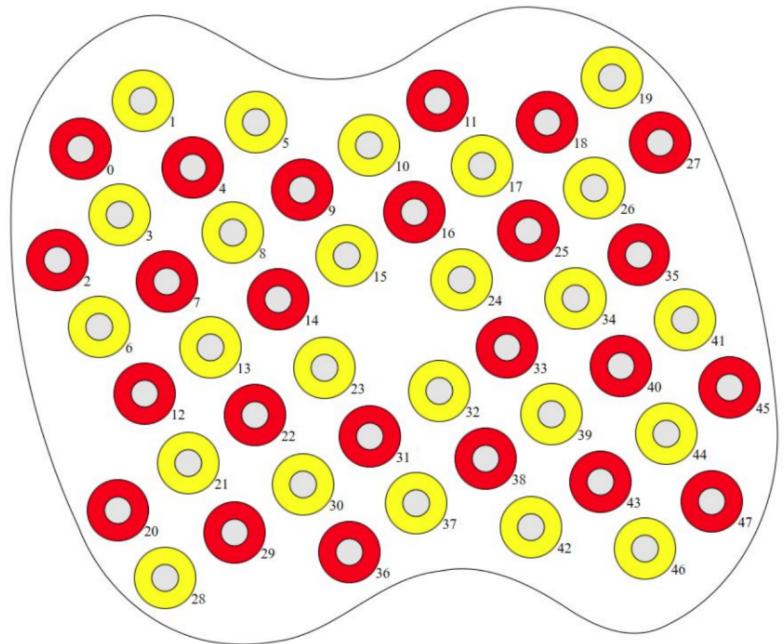
- “Avalam Bitaka”, Philippe Deweys (1995)
  - Réédité par Art Of Games (Vincent Sélenne, 2009)
- Ils seront présents aux soutenances !
- Nombreuses fois primé
  - As d’or du jeu de l’année à Cannes 1996
  - Jeu de l’année en Belgique et au Danemark
  - Mensa select USA 1998
  - ...
- Également utilisé dans un module similaire à l’UCL (Louvain La Neuve)





# Position initiale

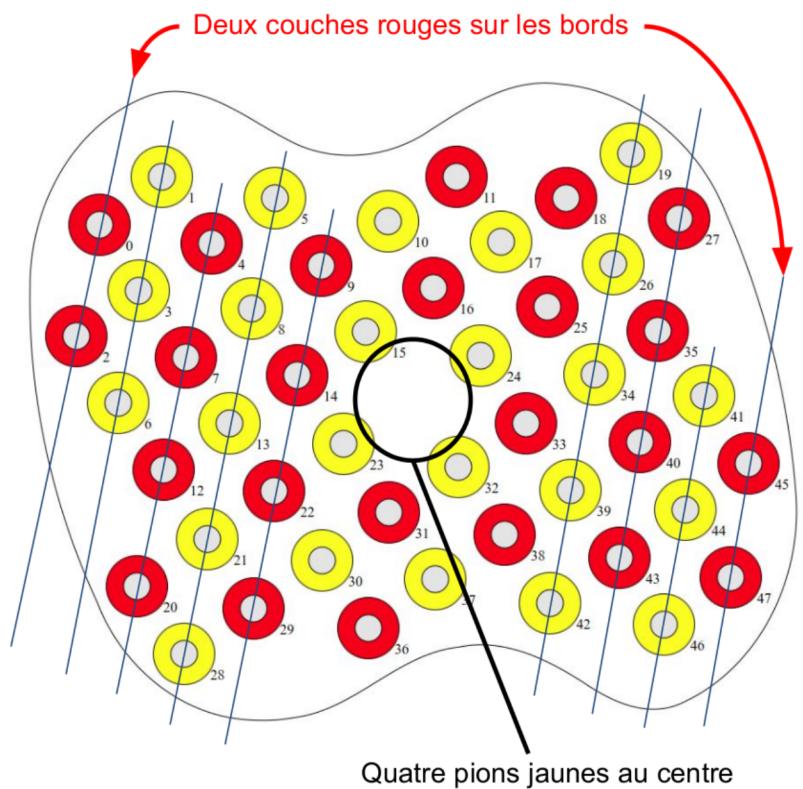
- 48 emplacements
- Deux couleurs
  - 24 jaunes, 24 rouges





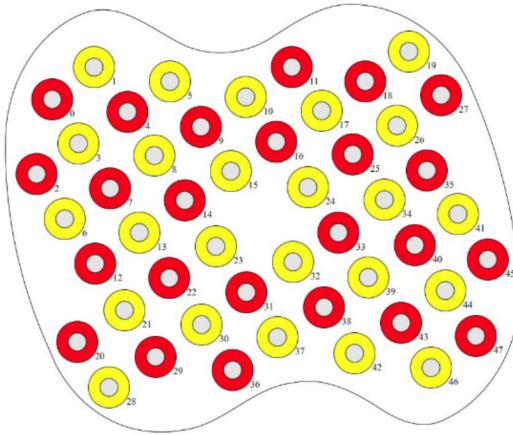
# Position initiale

- 48 emplacements
- Deux couleurs
  - 24 jaunes, 24 rouges
  - Répartis en couches





# Déplacements



- Les jaunes commencent
  - Car ils “semblent” désavantagés (ce qui reste à prouver)
- Chaque joueur joue à son tour
- Chacun peut déplacer TOUS les pions
- Un déplacement = TOUJOURS un empilement
  - Sur une case voisine (côtés ET diagonales)
  - Hauteur Max de la pile = 5



## Fin du jeu

- On joue jusqu'à ce qu'il ne reste plus aucun coup légal
  - Y compris si cela ne nous arrange pas !
- Le vainqueur est celui qui a le plus de colonnes dont le sommet porte sa couleur
  - Toutes les colonnes valent 1 point
- En cas d'égalité, les colonnes de 5 pions valent deux points

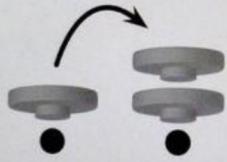
# Règles officielles

**AVALAM®**  
5<sup>TH</sup> EDITION



## Préparation du jeu

Disposer sur les trous les 48 pions en alternant les couleurs.



## Règles

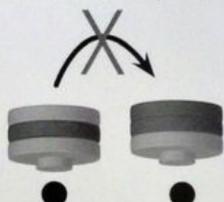
Chaque joueur choisit sa couleur.

Chaque joueur a le droit de jouer avec tous les pions, y compris ceux de l'adversaire.

## But du jeu

Posséder un maximum de tours dont le pion supérieur est de la couleur choisie.

Une tour est constituée de 1 à 5 pions. Jamais plus.



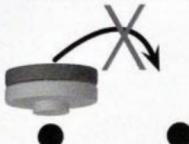
Un joueur est le propriétaire d'une tour lorsqu'un pion de sa couleur en occupe le sommet.  
Un pion isolé est également considéré comme une tour.



## Déplacements

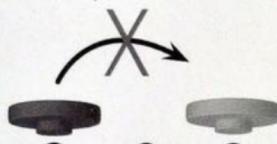
A tour de rôle, chaque joueur effectue un seul déplacement à partir de n'importe quel pion ou pile de pions. Chacun est donc libre de jouer où bon lui semble.

Un déplacement consiste à empiler le ou les pions sur un emplacement directement voisin (diagonale = ok) déjà occupé par un ou plusieurs pions (jamais sur un trou vide).

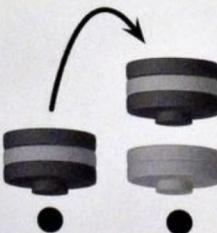


→ Chaque coup joué équivaut donc à libérer un emplacement.

→ Il est interdit de passer au-dessus d'un trou vide.



On déplace obligatoirement tous les pions situés sur un emplacement (de 1 à 4. Une tour de 5 pions ne se déplace pas).



Une pile de pions ne peut qu'augmenter, jamais diminuer.

Un pion ou une tour, isolés de tous les côtés, ne pourront donc plus bouger ni changer de propriétaire.

## Fin de la partie

La partie se poursuit tant qu'on peut effectuer des empilements. Le jeu se termine lorsque plus aucun déplacement n'est possible.

Le plateau n'est donc plus constitué que de tours ou de pions isolés.

Seul le dernier étage compte.

→ Le vainqueur est le joueur totalisant le plus grand nombre de pions isolés ou de sommets de tours de sa couleur.



→ Qu'une tour compte 1,2... ou 5 pions, elle vaut toujours UN point.

En cas d'égalité, c'est celui qui a le plus de tours de 5 pions (avec le pion de sa couleur au sommet) qui a gagné.



## *Ressources (cf. moodle)*

- **avalam-standalone**
  - **avalam-refresh**
  - **avalam-diag**
  - **libavalam**



## avalam-standalone.html

- Page Web permettant de jouer à deux joueurs
- Fonctionne sans serveur Web

=> Vous permet de **jouer partout** même sans avoir un jeu plateau de jeu d'avalam



# avalam-refresh.html

- Page Web permettant de visualiser périodiquement une position lue depuis un fichier JSON dans le répertoire courant
  - Fournie avec un exécutable en C permettant de jouer à deux, à réécrire
- A héberger **sur un serveur Web**

=> Vous permet d'offrir une **interface graphique** à vos programmes en C



# refresh : Format JSON

Encodage couleur :

- 0 : VIDE
- 1 : JAUNE
- 2 : ROUGE

```
{ "cols": [ "trait":2, {"nb":0, "couleur":0}, "scoreJ":23, {"nb":2, "couleur":2}, "scoreJ5":0, {"nb":1, "couleur":2}, "scoreR":24, {"nb":1, "couleur":1}, "scoreR5":0, ... ] }
```



## avalam-diag.html

- Page Web permettant de visualiser une position lue depuis un fichier JSON dans le répertoire courant, de lui attribuer une légende et une description
- A héberger **sur un serveur Web**

=> Vous permet de **documenter vos algorithmes** en présentant les positions étudiées



# diag : Format JSON

Encodage couleur :  
• 0 : VIDE  
• 1 : JAUNE  
• 2 : ROUGE

```
{ "cols": [ {"nb":1, "couleur":2}, {"nb":1, "couleur":1}, {"nb":1, "couleur":2}, {"nb":1, "couleur":1}, ... ]}
```



- Fonctions en langage C permettant la prise en main du jeu :
  - affichage des positions
  - génération des coups légaux
  - déplacement et mise à jour d'une position
  - calcul de score



# Etude du code fourni...

```
typedef struct {  
    octet nbJ;    octet nbJ5; // total et piles de 5  
    octet nbR;    octet nbR5;  
} T_Score;  
  
typedef struct {  
    octet nb;  
    octet cases[8]; // il peut y en avoir moins... ils vaudront 0  
} T_Voisins;  
  
typedef struct {  
    octet nb;  
    octet couleur; // couleur du sommet  
} T_Colonne;  
  
typedef struct { // Un tableau de colonnes  
    octet trait;  
    T_Colonne cols[NBCASES];  
} T_Position;  
  
typedef struct {  
    octet origine;  
    octet destination;  
} T_Coup;  
  
typedef struct {  
    int nb;  
    T_Coup coups[8*NBCASES];  
} T_ListeCoups;
```

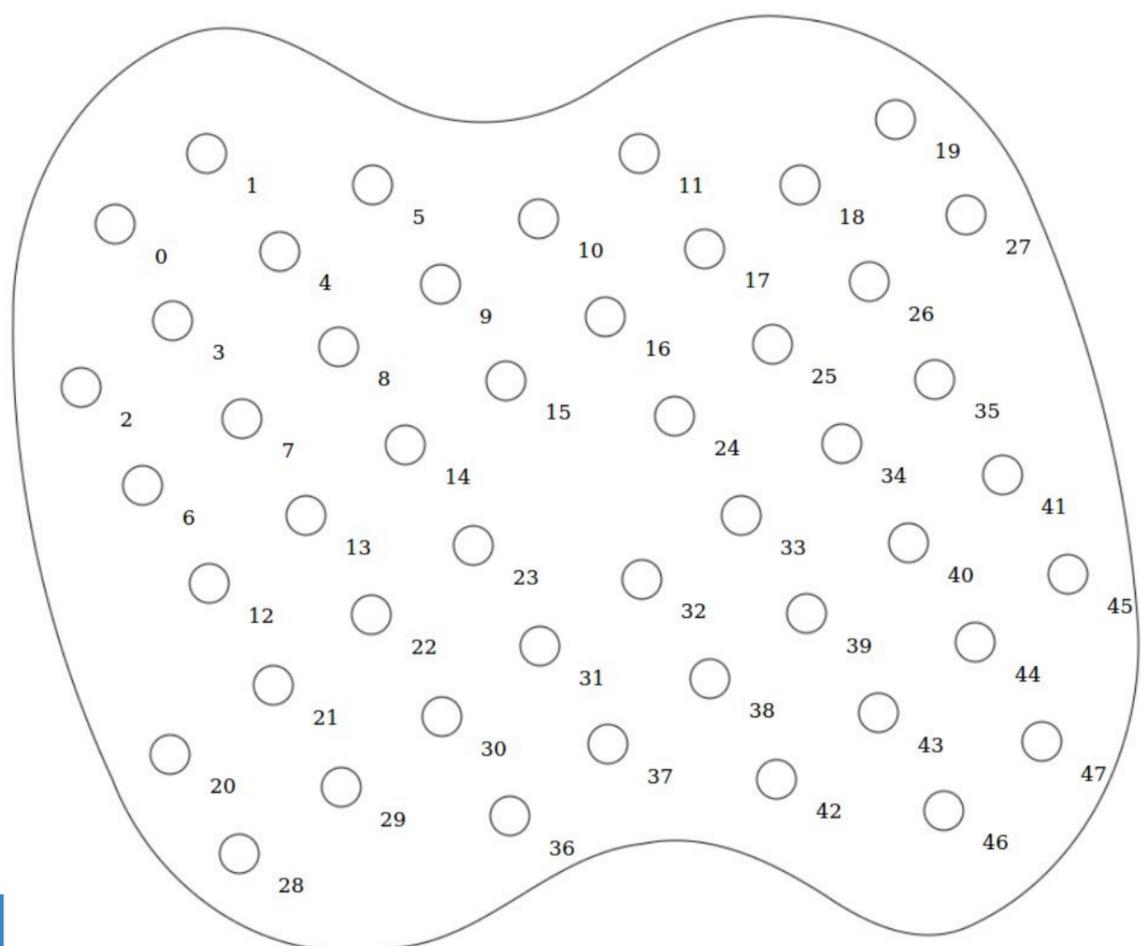


## Etude du code fourni

```
octet nbVoisins(octet numCase);  
T_Voisins getVoisins(octet numCase);  
void listerVoisins(octet numCase);  
T_Position getPositionInitiale();  
void afficherPosition(T_Position p);  
T_Position jouerCoup(T_Position p, octet origine, octet destination) ;  
T_ListeCoups getCoupsLegaux(T_Position p) ;  
void afficherListeCoups(T_ListeCoups l);  
T_Score evaluerScore(T_Position p);  
void afficherScore(T_Score s);
```



# Indices des cases



Challenge Moteur de jeux



## *Au Boulot !*

- Travail demandé
- Constitution des groupes



# Livrable 1 : Développement en C

- Deux programmes
  - standalone.exe
    - Jeu à deux joueurs, saisie des coups au clavier, affichage par avalam-refresh
  - diag.exe
    - Saisie de positions sur la ligne de commande, affichage par avalam-diag



# Cahier des charges standalone.exe

- Permet de jouer à deux joueurs
- Saisie des coups chacun son tour au clavier
- A chaque coup, produit un fichier json avec la position et le score
  - Ce fichier sera utilisé par avalam-standalone.html
  - Le nom du fichier doit avoir été passé en ligne de commandes
  - Un nom par défaut est prévu dans le programme
- Déetecte la fin de la partie et affiche le score à l'écran



# Cahier des charges diag.exe

- Permet de passer :
  - un numéro du diagramme
  - une position en ligne de commandes, type “FEN”
- Saisie du nom du fichier et d'une chaîne de description au clavier, facultatifs
- Produit un fichier json avec la position et la description utilisé par avalam-diag



# Format “FEN” Forsyth-Edwards Notation

- Description compacte d'une position d'échecs
  - rnbqkbnr/pppppppp/8/8/8/PPPPPPPP/RNBQKBNR w
- avalam-fen : utilisation des symboles
  - u d t q c pour les jaunes
  - U D T Q C pour les rouges
  - 0 1 ... n pour indiquer un nombre de colonnes vides
  - j/r pour le trait
  - en suivant l'ordre des indices des cases



# Livrable 2 : Stratégies de jeu



- Jouer, pour **découvrir** des situations intéressantes à partir desquelles construire une “théorie”
  - *Sérendipité*
- Développer ces algorithmes / tactiques / stratégies / plans permettant de jouer de bons coups
  - en profiter pour définir un vocabulaire
- Documenter ces stratégies dans un “recueil”
  - à l'aide des diagrammes générés par diag.exe



# Livrable 3 : Moteur de jeu

- Développement d'une fonction C qui sélectionne le prochain coup à jouer parmi les coups légaux
  - Temps Maximum : 30 secondes
  - Cette fonction permettra de produire un “moteur de jeu”
  - Un tournoi sera organisé pour confronter les moteurs de jeux entre eux
- Plus de détails sur cette fonction prochainement...



# *Planning*



# Echéances

- Constitution des groupes : équipes de 4
- Affectation d'un référent à chaque groupe



# Modalités d'évaluation

- Remise de documents
- Soutenances intermédiaires
- Soutenance finale : groupes finalistes



# *Conclusion & Perspectives*

- **Programmation des jeux & Intelligence Artificielle**



# Avalam : Caractéristiques

- Deux adversaires jouent à tour de rôle
- La situation globale du jeu est connue de chacun des joueurs : “**jeu à information complète**”
- La chance n’intervient pas : “**jeu déterministe**”
- Les gains d’un joueur représentent exactement les pertes de l’autre joueur : “**jeu à somme nulle** »

**Condition d’application  
de l’algorithme du “minimax”**



# Une brève histoire de ... la programmation des jeux

- 1769 : Le Turc, baron Von Kempelen
- 1928 : Théorème du minimax, John Von Neumann
- 1956 : Elagage alpha-bêta par John McCarthy
- 1997 : Deep Blue bat Kasparov
- 2016 : AlphaGo bat Lee Seedol
- 2017 : AlphaZero bat Stockfish





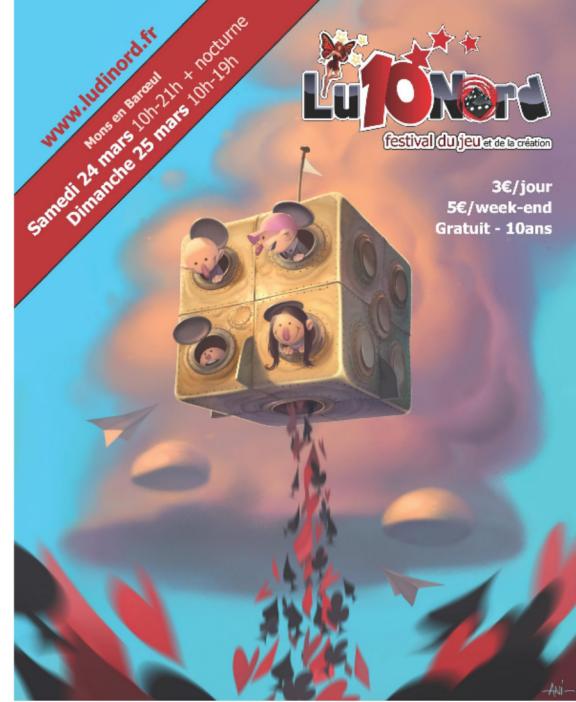
# Autour de cette activité

- Une asso de jeux de société à 2i ?
  - Cf. “Centrale Games”
- Un tournoi d’étudiants ?
  - le meilleur rencontre le moteur vainqueur
- Ludinord
- Projet Recherche à Centrale :
  - “Alphalam : deep learning pour avalam”



# Ludinord a 10 ans !

- 24 & 25 mars 2018
  - Mons en Baroeul (Fort de Mons)
- Cf. Sylvain Machen
  - Promo 2006
  - Président de la Fédéjeux
  - Ancien coordinateur Ludinord
    - Resp. Com en 2018



Joueuse Mons en Baroeul méo elopement Furet furet.com Transpolis Le Jouet GRAND METROPOLE

