

Mini-projet langage C

.....

Réalisation d'un jeu
Codage de joueur artificiel

Contraintes

- Développement du jeu par 4 personnes
 - Répartis sous la forme de 2 binômes
- Chaque binôme a la responsabilité du développement d'une partie
 - Interface graphique
 - Stratégies de joueur artificielle et chargement de cette stratégie

Contexte

- Nom du jeu...

Patience...

Binôme *Interface graphique*

.....

- Codage interface graphique
 - GUI/algorithmique
- Développé obligatoirement avec la librairie SDL (www.libsdl.org)
 - Librairie portable
 - Facile d'utilisation
 - Nombreux tutoriels sur Internet
 - Gain de temps dans le développement d'interfaces graphique simples
- Codage de l'arbitre
 - Vérification de la validité d'un coup
- Pour compiler sous Linux

```
gcc -o executable fichier1.c fichier2.c fichier3.c ... `sdl-config --cflags --libs`
```

Binôme *Interface graphique*

.....

- Doit permettre à zéro, un ou deux joueurs humains de s'affronter
 - Nombre de joueurs = donnée du jeu
 - Lorsqu'un joueur n'est pas présent, une stratégie artificielle manipule le joueur virtuel
 - But du 2^{ème} binôme
- Au minimum, doit pouvoir être lancé entièrement en ligne de commande
 - Choix du nombre de joueur
 - Indication des librairies de stratégie à utiliser (voir partie binôme 2)
 - Ligne de commande de la forme

```
Game 1 ./strategy/simple.so
```
- Doit pouvoir ressortir les résultats de la partie dans un fichier texte
 - En ajout (ne pas écraser les précédentes exécutions)
 - Selon un affichage simple qu'il est possible de retraiter automatiquement par la suite
 - Séparation par des tabulations par ex, ...

Contexte

.....

- Nom du jeu...

Patience... encore !

Binôme *Stratégie de jeu*

.....

- Code une stratégie de jeu de joueur artificiel
 - Algorithmique/système
- Sous forme d'une librairie dynamique
 - À fournir à l'exécution du jeu
 - Doit respecter les règles
 - En cas d'erreur, le coup est annulé -> passage au joueur suivant
 - Si 3 erreurs dans une partie -> joueur éliminé (perdu !)
- Doit fournir le code du chargement de la librairie
 - Chargement explicite
 - Libération propre des ressources

Contexte

- Nom du jeu...
 - Stratego
 - Jeu de société de stratégie et de bluff
 - Se joue à 2 joueurs
 - Tour à tour
- Caractéristiques principales
 - Sur grille de 10*10 cases
 - 40 pions par joueurs
 - Caractéristiques différentes par pions
 - Type de pions inconnu de l'adversaire
- Règles complètes en pdf sur madoc

Stratego



Binômes *Interface graphique + Stratégie de jeu*

.....

- Besoin de définir une API de dialogue entre l'exécutable et la librairie
 - Cette API doit être commune à l'ensemble des binômes
- Décision à prendre collectivement pour la semaine prochaine
 - Deadline **mercredi 04/12/2013 à 17h15**
 - Utiliser le forum du cours (partie projet) sur madoc
- Me faire une ou plusieurs propositions (max 3)
 - Je prends la décision finale
 - Tout le monde utilise cette interface

Évaluation

.....

- Évaluation par binôme
 - Mais tient compte du travail commun du groupe de 4
- Prends en compte
 - Fonctionnement général de l'application (plantage, ...)
 - La qualité du code (gestion mémoire, ...)
 - La propreté du code (commentaires, indentation, ...)
- Note *a priori* basée
 - 1/3 sur le travail du groupe de 4
 - 2/3 par binôme
- Peut-être (en fonction du résultat)
 - Réalisation d'une compétition entre toutes les stratégies
 - Les vainqueurs auront un bonus
 - Pas de points négatifs par ce biais