

Projet application graphique
Y. Stroppa
PSB 2020

Sujets possibles de projet et détail de démarrage

Choix parmi les sujets proposés :

- 1/ Ane rouge
- 2/ rush hour
- 3/ Taquin numérique
- 4/ Taquin image
- 5/ Fiver jeu de positionnement de croix
- 6/ Kakuro
- 7/ Fubuki
- 8/ Suduko diagonal
- 9/ Suduko hyper
- 10/ Suduko argyle
- 11/ Suduko non consécutif
- 12/ Sudoku irrégulier
- 13/ Sudoku untouch
- 14/ Binero
- 15/ Cube3D représentation 2D
- 16/ Cube3D représentattion 3D
- 17/ Jeu de puzzle Gagne ton Papa --multi-joueurs
- 19/ Koridor -- stratégie --multi-joueurs
- 20/ Reconstitution de photos
- 21/ Puissance 4

A construire une application en Javascript et un rapport d'élaboration en binôme ou Trinome.

Description des réalisations à produire :

- 1/ Jeu de l'âne rouge grille de dimension 4*5

Il est constitué de plusieurs blocs que l'on peut déplacer. L'idée de base est de déplacer le bloc rouge situé en haut dans sa configuration de base vers le bas au centre. Ce déplacement peut se faire qu'en essayant de lui faire de la place à partir des autres pièces. Les mouvements possibles sont à droite, à gauche en haut et en bas

- 2/ Rush hour grille de dimension 6*6 : 4 niveaux possibles : Débutant/Intermédiaire/Expert/Advanced

C'est un jeu qui consiste dans un encombrement de voitures de dégager une des voitures (la voiture rouge) vers la sortie. Le tout est possible en déplaçant les autres véhicules pour lui faire de la place. Chaque véhicule ne peut se déplacer que sur sa colonne ou ligne respectivement. Les véhicules sont de longueur différentes et leur mouvement n'est possible que sur leur axe.

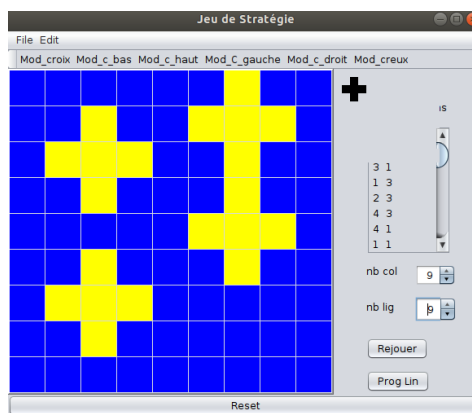
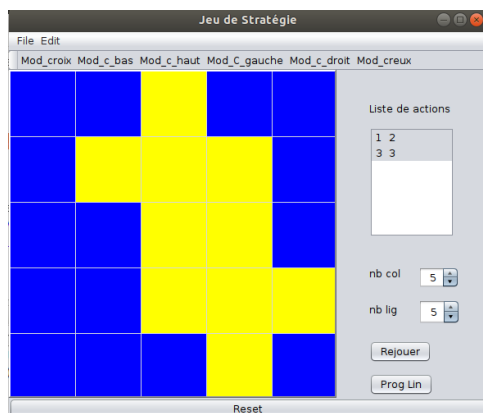
- 3/ Le jeu du Taquin Numérique grille de dimension 4/4

C'est un ensemble de plusieurs cases numérotés de 1 à jusqu'à 15 que l'on doit remettre dans l'ordre par mouvements successifs des cases car on dispose que d'une seule case vide et des possibilités de mouvements dans les quatre directions.

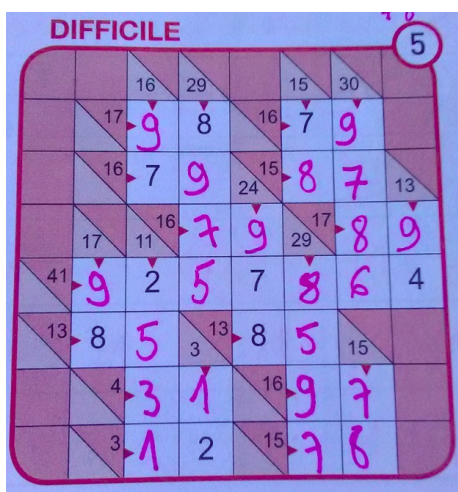
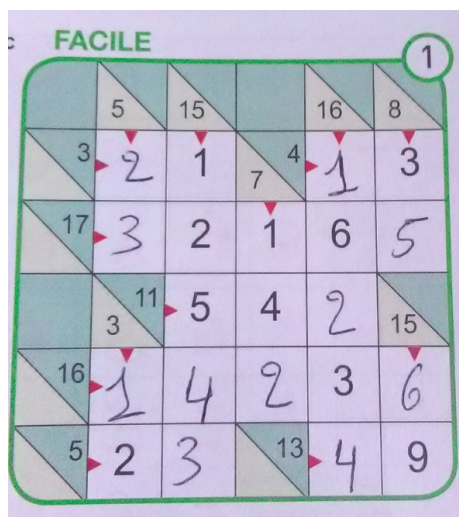
4/ Le jeu du Taquin Photo : grille de dimension 4/4

C'est un ensemble de plusieurs cases dans lesquelles on va retrouver une partie d'une photo que l'on doit remettre dans l'ordre par mouvements successifs des cases car on dispose que d'une seule case vide et des possibilités de mouvements dans les quatre directions.

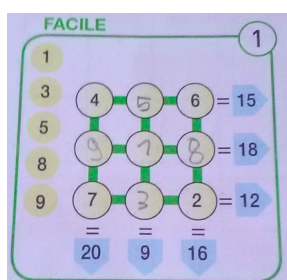
5/ Fiver : c'est une grille de dimension $n \times n$ dans laquelle l'utilisateur à partir d'un motif donné (une croix) doit remplir de la même couleur toute la grille



6/ Kakuro : jeu qui consiste à compléter les cases vides en fonction des sommes à réaliser (indications fournies). Trois niveaux possibles : Facile/Moyen/Difficile



7/ Jeu du Fubuki : identique au précédent mais avec des indications sur les chiffres à positionnés pour effectuer des sommes à retrouver. Quatre niveaux possibles Facile/Moyen/Expert/Diabolique



8/ à 13/ ce sont des variantes du sudoku grille de dimension 9x9

8/sudoku diagonal : chacune des diagonales bleues est également sudoku

9/sudoku hyper : chacune des zones bleues est également zone sudoku.

10/ sudoku argile : tous les chiffres situés sur une même diagonales en pointillées doivent être différents.

11/ sudoku non consécutif : En +, deux cases horizontalement ou verticalement ne peuvent contenir de chiffres consécutifs.

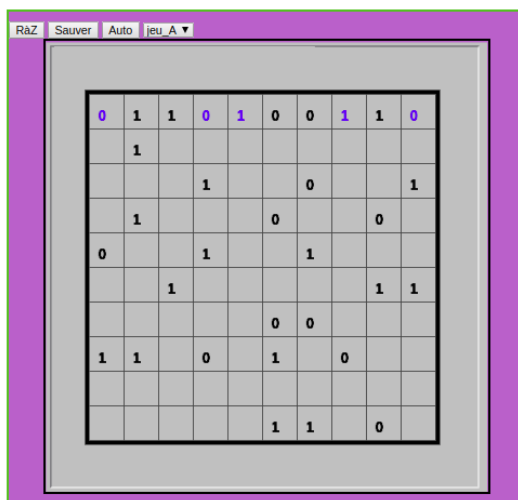
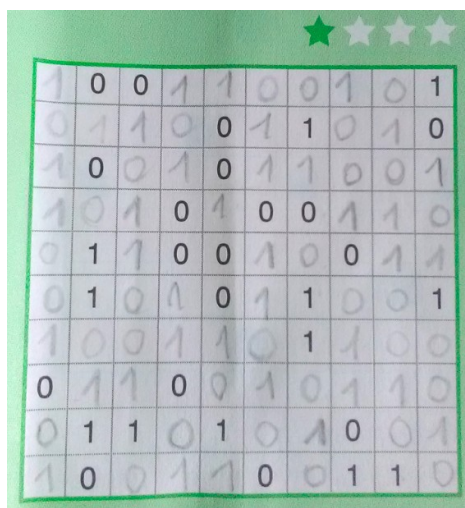
12/ sudoku irrégulier : Les blocs n,e sont plus des carrés mais des formes aléatoires.

13/ sudoku untouch : deux chiffres identiques ne peuvent se toucher en diagonale.

14/ Binero grille de dimension 10x10

Chaque ligne et chaque colonne doit contenir le même nombre de 0 et de 1. On ne peut pas placer plus de deux 0 ou de deux 1 côte à côte ou l'un en dessous de l'autre. Deux colonnes et deux lignes ne peuvent être identiques.

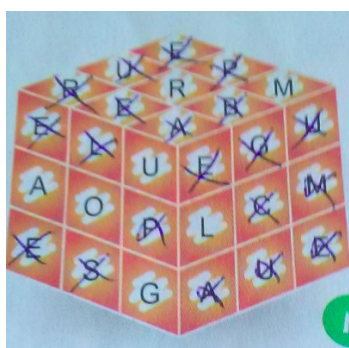
Jeu Binero 2020 V2



Pour le développement de ce jeu, on pourra prévoir plusieurs interfaces possibles avec une zone de saisie pour les valeurs ou un changement de valeur par simple clique sur la zone. De plus des vérifications systématiques des règles devront permettre de vérifier la solution proposée et avertir l'utilisateur des erreurs. Par des petites flèches le long des lignes ou colonnes qui posent problèmes.

15/ Cube 3D : représentation 2D

L'idée est un cube composé de différentes cases dans lesquelles on va retrouver des lettres à partir desquelles il faut reconstruire plusieurs mots en fonction de la définition donnée.



16/Cube 3D : représentation 3D en partie SVG

Réel cube 3D avec possibilité de le tourner dans l'espace pour déterminer des mots en fonction des définitions fournies

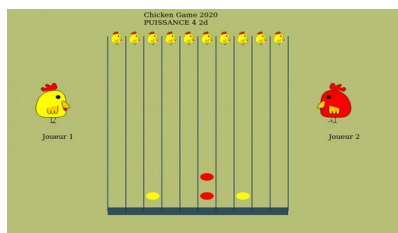
17/ Jeu du corridor

Jeu qui se joue à deux, 4 et éventuellement contre la machine ...(IA) consiste à se déplacer vers le côté opposé et de bloquer l'avancée de l'adversaire en posant des briques sur une grille de dimension donnée.

18/Jeu gagne ton papa : le jeu se joue à un joueur et à deux joueurs avec défis de reconstitution d'un puzzle à partir de brique de forme déterminer dans un espace de dimension donné

19/ Affichage de photo à reconstituer de dimension variable et de difficultés variables dans un temps donné

20/ Jeu Puissance 4 : svg



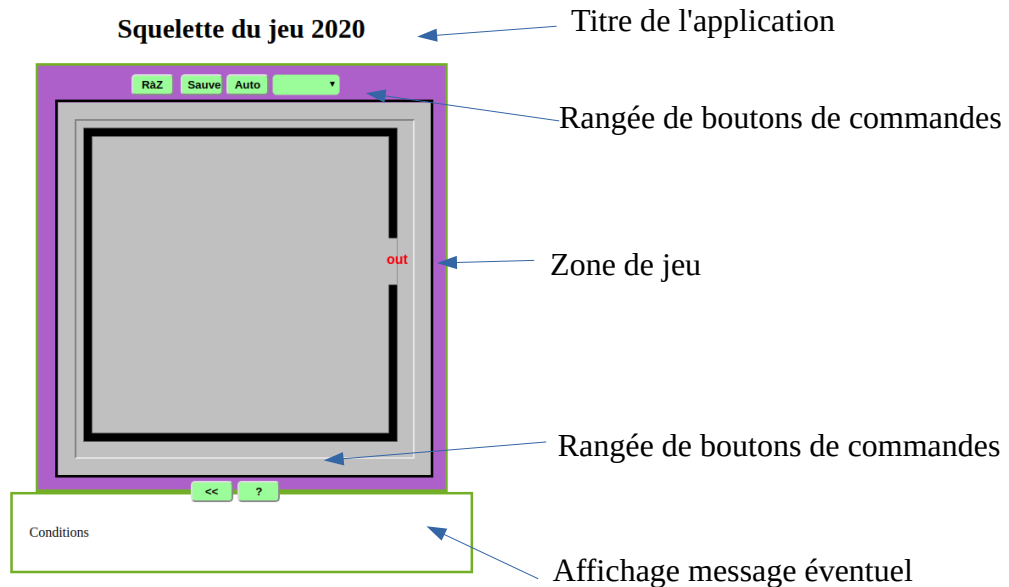
Voir le développement de la partie IA

Construire le fichier html et css correspondant à l'application

Le fichier html ne doit contenir que la définition des éléments d'affichage tels que :

Une rangée de boutons de commandes en haut et en bas
le cadre de travail == canvas ou svg

Comme représenté ci-dessous, on ajoutera en inclusion les fichiers nécessaires



La dimension de la zone de jeu doit permettre l'affichage du message de réussite à la fin de la partie si le joueur a gagné.

Veuillez générer les fichiers html et css pour effectuer l'affichage de cette disposition.

Générez un script nommé comportement.js dans lequel on pourra écrire les différents comportements souhaités pour cette application.

On commencera par initialiser un ensemble de variable pour récupérer un descripteur sur la zone canvas afin de dessiner dessus.

```
var nbpieces=10;
var fenetre = document.getElementById('canvas');
var hh=canvas.getContext('2d');
hh.canvas.style.border="3px solid #000";
hh.translate(0.5,0.5);
trace();
```

Définir la fonction trace() qu devra permettre de dessiner le cadre et les différents contours nécessaires pour le jeu.

On souhaite que le cadre soit dessiné en fonction de la taille de la "matrice" à représenter.

```
var nb_lignes=6 ;
```

```
var nb_colonnes=6 ; la largeur d'une case sera une unité nommée var a=56.
```

La fonction trace() ressemblera à

dessiner les différents cadres pour effectuer l'effet attendu en utilisant la variable hh et ses fonctions font, textAlign, fillStyle, fillRect

On regardera les différentes définitions de ces fonctions pour pouvoir les implémenter

On prendra garde également à l'ordre d'affichage des éléments pour permettre la superposition des différents affichages

```
}
```

Codage du jeu

Maintenant que l'on a le cadre de notre application, il faut se soucier du contenu de la zone de dessin ... ceci va dépendre des éléments que l'on doit rajouter et du déplacement que l'on autorise à effectuer.

Plusieurs choses sont à effectuer avant d'implémenter cette partie :

Comment code-t-on la position des pièces sur ce maillage et leur forme ?

Des pièces de deux cellules, de trois cellules

Des pièces verticales ou horizontales

Comment sont stockées ces informations et dans quoi ?

Comment va-t-on simuler le mouvement d'une pièce d'une case vers une autre case ? et quels sont les événements qu'il va falloir gérer pour effectuer cette actionon regardera mousedown, mouseup, mousemove

Pour tracer un objet dans la zone graphique on pourra s'appuyer sur la fonction

```
function drawLine(xi,yi,xf,yf){  
    hh.beginPath();  
    hh.moveTo(xi,yi);  
    hh.lineTo(xf,yf);  
    hh.stroke();  
}
```

Question subsidiaire : dans le cadre de la programmation d'un générateur de jeu automatique comment peut-on vérifier que la combinaison que l'on propose à l'utilisateur a bien une solution ???