

0	1	1	0
0	1	0	1
1	0	1	0
1	0	0	1

TAKUZU

Rapport du projet

Table des matières :

- Objectifs du projet
- Organisation du programme
- Fonctions
- Difficultés rencontrées
- Manuel de l'utilisateur
- Organisation / répartition du travail

Objectifs du projet

Pour aboutir au projet du fameux « Takuzu », nous avons pour objectifs :

- télécharger un puzzle depuis un fichier ;
- permettre à l'utilisateur d'essayer de résoudre le puzzle en mode terminal ;
- permettre à l'utilisateur d'essayer de résoudre le puzzle en mode graphique ;
- permettre à l'utilisateur d'annuler, un par un, les coups précédents ;
- Le confort d'utilisation est un élément important :
 - Il doit être possible de remplacer en mode graphique les chiffres 0 et 1 par des jetons blancs et noirs ou des ronds et des croix, etc. ;
 - Donner lorsqu'il le demande un indice à l'utilisateur, en essayant d'appliquer une des trois règles du jeu.

Organisation du programme

Notre programme est organisé en deux parties (comme nous l'avons dit dans les « Objectifs ») : en mode terminal et en mode graphique. C'est-à-dire que l'utilisateur peut choisir de jouer au Takuzu sur le terminal : le puzzle est affiché à chaque tour et l'utilisateur doit entrer au clavier les coordonnées d'une case. Si cette case est déjà occupée, son contenu est supprimé, si elle est libre, le programme demande à l'utilisateur la valeur qu'il souhaite y stocker.

Le programme doit empêcher les coups illégaux et afficher un message lorsque l'utilisateur a gagné le puzzle.

Mais il peut aussi choisir de jouer sur le mode graphique : la grille est affichée dans une fenêtre, un clic sur une case change sa valeur de façon cyclique entre 1, 0, et vide.

Toutes les valeurs initiales du programme sont mises dans une même fonction pour bien rendre le code lisible sans avoir à trier entre valeur et fonction.

Fonctions

Pour lancer le programme, nous l'avons incrémenté de plusieurs fonctions. Ces fonctions sont plus ou moins importantes pour le lancement du jeu.

Voici les fonctions principales du Takuzu :

Pour le mode terminal :

- **LOADFILE_choix(choix)** : selon le choix de l'utilisateur, les différents puzzle sont chargés sous la forme de listes de listes dans le terminal grâce à l'utilisation de la fonction **LOADFILE(fileName)**
- **MAJ(i,j,n,choix)** : on définit les coordonnées des différents éléments des listes puis on affiche les puzzle sous forme de listes de liste grâce à la fonction 'affiche' , Par exemple si on choisit le niveau débutant (choix==2), alors la variable n sera affichée aux coordonnées[i][j] dans le puzzle de taille 4.
- **compare_listes(lst1,lst2)** : on vérifie que les listes correspondant au puzzle et à sa correction on la même taille puis si elles contiennent les mêmes valeurs.
- **verifier()** : de par la fonction **compare_listes(lst1,lst2)**, on peut déterminer si le puzzle en cours est identique à sa solution(nous avons présenté des puzzles qui n'ont qu'une réponse possible), si c'est le cas, alors l'utilisateur gagne la partie.
- **bloquer(g)** : on prend les valeurs initiales des listes des puzzles, des tuples sont utilisés .Si la valeur correspondant à des coordonnées n'est pas vide(un tiret), alors on bloque le tuple(les coordonnées et la valeur correspondant) afin de ne pas endommager le puzzle initial.

Pour le mode graphique :

- `clic_4x4(event,cycle)` : elle sert à valider un clic qui peut modifier la valeur de `lst[i][j]` (en 1,0 ou vide) de la grille de taille 4 avec un clic sur la case qui le représente, les cases où l'on peut rajouter une valeur sont sélectionner pour que celle qui ont déjà une valeur ne peuvent pas être mutable par le joueur et donc on peut toujours garder la grille initiale.
(Même principe pour `clic_6x6(event,cycle)` et `clic_8x8(event,cycle)`)
- `grille_4x4()` : elle contient une liste de forme `lst[i][j]` pour 4 listes de 4 éléments, qui sera utiliser pour appliquer chaque élément de la liste dans une seule case par rapport à son indice.
(Même principe pour `grille_6x6()` et `grille_8x8()`)
- `verifier()`: c'est une fonction qui regroupe toutes les conditions du jeu Takuzu que lorsqu'elles sont vraies pour que la grille présentée par le joueur soit correcte, lorsque celle-ci l'est, une fenêtre s'ouvre pour informer le joueur qu'il a résolu la grille
- `grille(n)`: elle génère une grille en couleur de taille n

Difficultés rencontrées

En réalisant se projet, nous avons rencontré plusieurs difficultés :

- La première difficulté qui nous a posé problème est le fait de réussir à afficher la liste de liste initiale avec le premier élément au centre de la première case, le deuxième élément au centre de la deuxième case, ..., jusqu'au dernier élément placé au centre de la dernière case. Pour y remédier, nous avons fait une boucle qui en fonction de la hauteur et largeur de la case et de l'indice de l'élément (ici liste) affiche sa valeur. La valeur correspond ensuite à la liste de la liste, celle-ci affiche ses éléments séparés d'une longueur identique : on voit apparaître des 0, 1 et vide à leur emplacement respectifs.
- La difficulté suivante concerne le fait d'appliquer l'une des règles du Takuzu, il ne doit pas y avoir plus de deux 0 ou 1 placés l'un à coté ou en dessous de l'autre. Pour résoudre ce problème, nous avons réfléchi à plusieurs idées qui ont été fructueuse mais deux sont sorties du lot :
 1. Pour le mode terminal, nous avons comparé la liste correspondant au puzzle à celle correspondant à la solution. Lors de nos premiers essais, nous avons utilisé le module numpy et la classe array pour former des matrices et ce mode de fonctionnement marchait très bien. Malheureusement les valeurs initiales des grilles ne pouvaient être bloquées et l'utilisateur pouvait les modifier donc nous avons tout supprimé puis simplement utilisé des listes de liste et des tuples afin de bloquer les valeurs initiales de la grille. Cependant un problème persiste, la comparaison des listes ne s'effectue pas malgré de nombreuses vérifications sur les fonctions utilisées.

2. Pour le mode graphique, nous avons pu mettre une fonction qui contient deux paramètres : la liste et la valeur (0 ou 1). Elle vérifie sur la liste utilisée si il y a ou non trois valeurs identiques affichées à la suite, si c'est le cas le joueur sera informé que sa grille contient une erreur.
- En mode graphique, réussir à afficher une valeur dans la grille au bon endroit. Au tout départ, nous avons réussi à afficher une valeur au hasard lorsque l'on clique cependant ce n'était pas le résultat voulu : l'endroit où l'on avait cliqué affichait une valeur, donc lorsqu'on cliquait partout sur la fenêtre cela mettait plein de valeurs qui pouvaient se même se superposer sans même effacer la première valeur présente. Ensuite nous avons réussi à trouver une méthode : nous avons pris les coordonnées de l'emplacement du curseur lors du clic et les avons encadrées par les coordonnées possibles en x et en y de chaque case vide de chaque grille, en fonction du clic (gauche, droit ou pression sur roulette) cela affichera soit un 1, un 0 ou un ' ' (vide) au centre et modifiera aussi l'élément de la liste qui a été choisi en la nouvelle valeur.

Manuel de l'utilisateur

Mode terminal :

- Table de choix :
- Taper 0 : Règle du jeu
- Taper 1 : But du jeu
- Taper 2 : Niveau Débutant
- Taper 3 : Niveau Intermédiaire
- Taper 4 : Niveau Expert
- Taper q : Quitter
- Choisissez un niveau (Débutant, Intermédiaire, Expert), le puzzle s'affiche. Ensuite choisissez les coordonnées de l'emplacement à modifier (n° ligne puis n° colonne), puis, il vous est proposé d'entrée soit la valeur 1 soit la valeur 0, recommencez jusqu'à avoir remplis la grille.
- Attention, pour les coordonnées, si on veut la ligne 1, on tape 0, ligne 2 on tape 1, etc... de même pour les colonnes. Dès que le puzzle est correctement rempli, l'utilisateur gagné, on devrait voir affiché « Vous avez réussi »

Mode graphique :

- Cliquer sur « But du jeu » pour afficher le but
- Cliquer sur « Règles du jeu » vous affichera les règles qui sont à prendre en compte pour jouer au Takuzu
- Cliquer sur « Niveau » vous propose de choisir la difficulté (Débutant, Intermédiaire, Expert)
- Cliquer sur « Comment jouer ? » vous donnera une explication des touches nécessaire pour afficher une valeur dans une case
- A partir du moment où le joueur choisit un niveau, le jeu est en route :
 - Faites un clic gauche sur la souris pour afficher 1 dans la case voulu
 - Faites un clic droit sur la souris pour afficher 0 dans la case voulu

- Faites un clic sur la roulette de la souris pour afficher une case de nouveau vide

Une fois le puzzle terminé, le joueur clique sur « Vérifier » pour voir un message s'afficher :

- en cas de victoire : « Félicitation vous avez réussi ! »
- si la grille à au moins une erreur : « Attention ta grille est incorrecte ! », cela indique qu'il faut revenir sur la grille pour la retravaillé jusqu'à arriver à sa solution

Organisation / répartition du travail

Pour la répartition du travail, nous avons décidé de ce mettre chacune de notre côté, pour ensuite réunir les idées trouver et les faire partager à son binôme pour voir si cela peut-être utilisé, transformer ou même ne pas être validé car on sait que l'on peut trouver mieux.

De plus, nous mettons nos programmes à exécution pour aider à corriger un bug et/ou une erreur non vu par celle qui a écrit le programme.

Nous avons écrit deux programmes, l'un pour jouer en mode terminal et l'autre pour le mode graphique. Ainsi, à l'ouverture du dossier l'utilisateur choisi quel mode lancer en ouvrant un des deux fichier.