



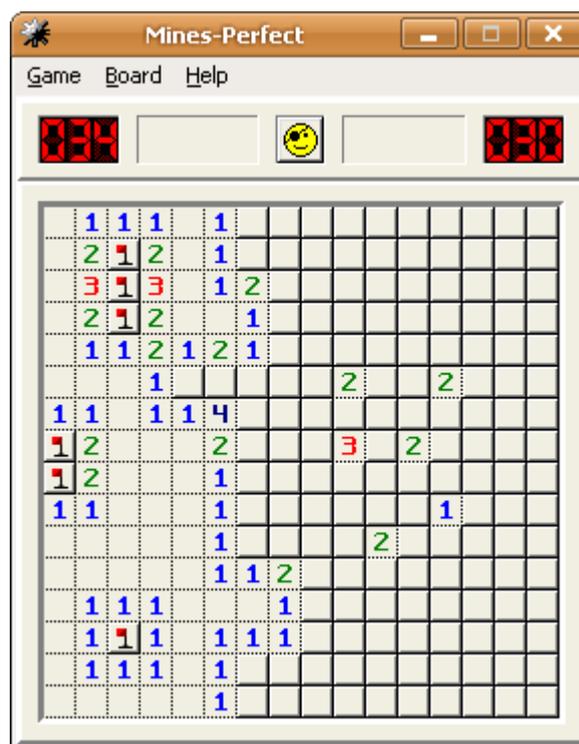
Le démineur

I- Présentation

Définition wikipédia :

[https://fr.wikipedia.org/wiki/D%C3%A9mineur_\(genre_de_jeu_vid%C3%A9o\)](https://fr.wikipedia.org/wiki/D%C3%A9mineur_(genre_de_jeu_vid%C3%A9o))

Le Démineur (Minesweeper) est un jeu vidéo de réflexion dont le but est de localiser des mines cachées dans une grille représentant un champ de mines virtuel, avec pour seule indication le nombre de mines dans les zones adjacentes.



De nombreuses versions gratuites du Démineur existent, mais la plus connue est celle fournie par défaut avec le système d'exploitation Microsoft Windows. Cette version particulière, à défaut d'inventer le concept, l'a popularisé, et la désignation « Démineur » identifie souvent cette version précise et non le principe général du jeu.

Vous pouvez tester le fonctionnement du jeu ici :

<https://xn--dmineur-bya.eu/>

Chaque case de la grille peut soit cacher une mine, soit être vide. Le but du jeu est de découvrir toutes les cases libres sans faire exploser les mines, c'est-à-dire sans cliquer sur les cases qui les dissimulent.

Lorsque le joueur clique sur une case libre comportant au moins une mine dans l'une de ses cases avoisinantes, un chiffre apparaît, indiquant ce nombre de mines. Si en revanche toutes les cases adjacentes sont vides, une case vide est affichée et la même opération est répétée sur ces cases, et ce jusqu'à ce que la zone vide soit entièrement délimitée par des chiffres. En comparant les différentes informations récoltées, le joueur peut ainsi progresser dans le déminage du terrain. S'il se trompe et clique sur une mine, il a perdu.

On peut signaler les cases contenant des mines présumées par un drapeau en cliquant sur le bouton droit de la souris – mais ce n'est aucunement obligatoire. Il faut faire attention à ne pas signaler une case saine par un drapeau, car cela peut induire en erreur ; ce n'est toutefois pas aussi pénalisant que de découvrir une mine.

II- L'interface graphique version 1

On vous fournit deux versions de cette interface graphique. La version 1 est la plus simple, elle n'inclut pas les fonctionnalités de chronométrage de la partie qui sont assez difficile à implémenter hors de la classe GUI. C'est pour cela que la version 2 les intègre. A part cette différence, les deux versions sont parfaitement identiques, vous pouvez commencer avec la 1 puis basculer sur la 2 pour obtenir les fonctionnalités supplémentaires à tout moment. Si le répertoire GUI_demineur est dans le même répertoire que votre fichier :

Vous devez importer la classe GUIdemineur depuis le module fourni :

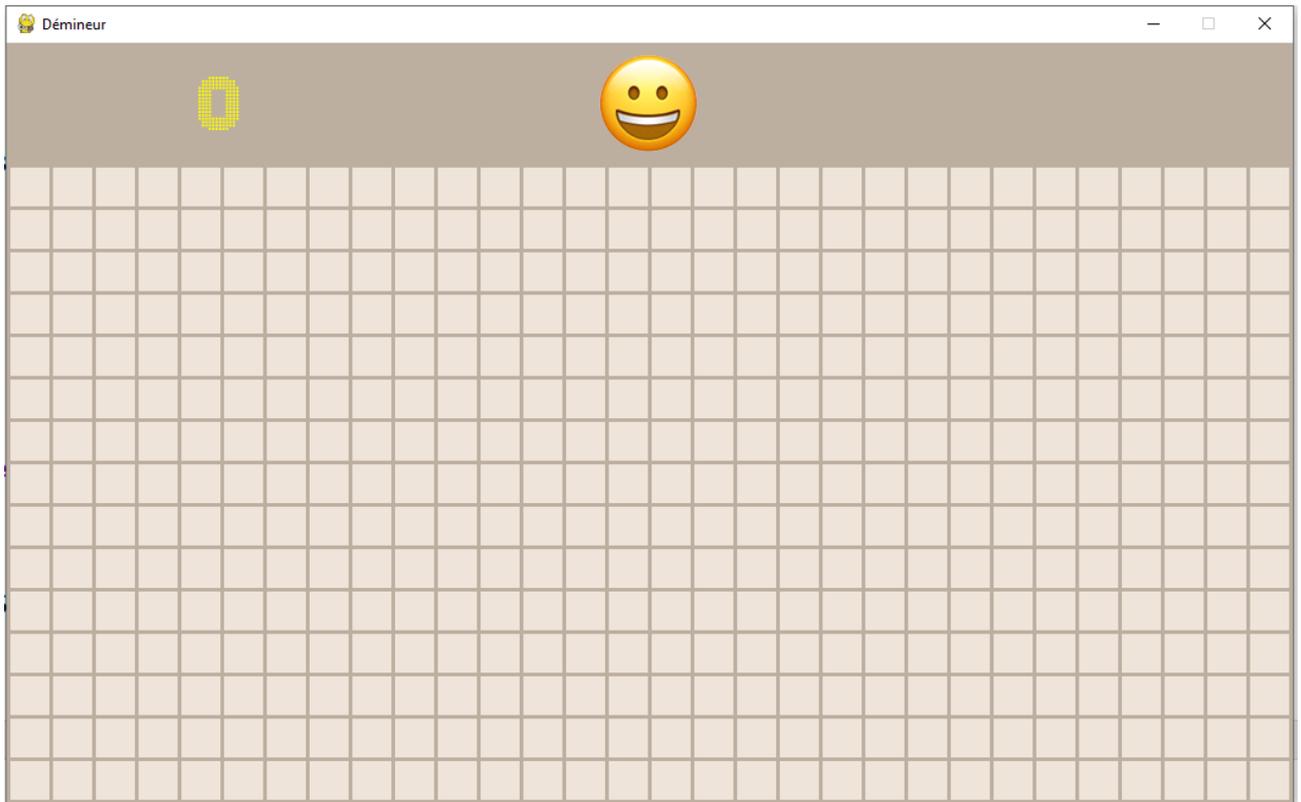
```
from GUI_demineur.guiDemineur_V1 import GUIdemineur
```

Constructeur de la classe :

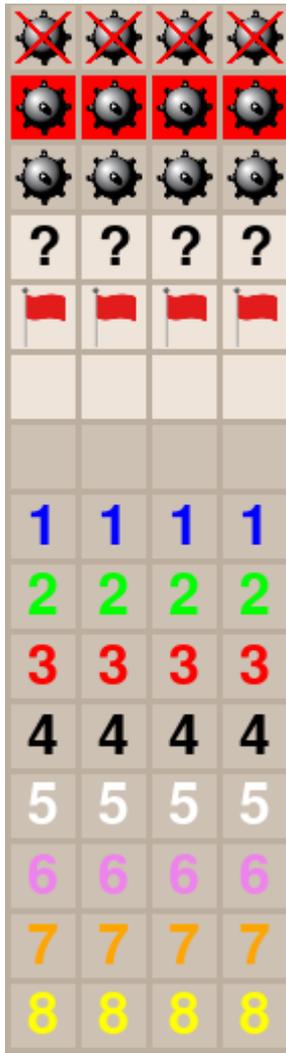
- Attribut public : **aucun**.
- Le constructeur prend en paramètre :
 - **nb** : le nombre de case sur la largeur de la grille. La hauteur de la grille est automatiquement deux fois plus petite que sa largeur.
 - **w** : La taille en pixel d'une case.

Par exemple pour une instancier grille de 30 cases par 15 de dimension 32 pixels de coté :

`GUI = GUIdemineur(30, 32)`



- Les méthodes :
 - **refresh(g, mine , emo)** : Cette méthode rafraichie l'affichage du plateau de démineur conformément à la grille passée en argument.
 - **g** est une liste de nb listes (ou n'importe quel indexable contenant des éléments itérables). Par exemple une grille de 30*15 :
 - `g[0][0]` est la case en haut à gauche ;
 - `g[0][29]` est la case en haut à droite,
 - `g[14][29]` est la case en bas à droite.



Attention : Le contenu de la grille définit le dessin de la case :

- -6 -> Bombe barrée : case faussement attribuée à un bombe
 - -5 -> Bombe sur fond rouge : Explosion, c'est perdu
 - -4 -> Bombe découverte
 - -3 -> ? sur case cachée -> vous avez un doute
 - -2 -> Drapeau rouge sur case cachée : vous pensez que c'est une bombe, le click devra être bloqué
 - -1 -> Case cachée
 - 0 -> case découverte vierge
 - 1 à 8 -> case découverte avec 1 à 8 bombe dans les cases adjacentes
- **mine** correspond au nombre de mine à deviner, à afficher à gauche du bandeau. Attention, il faut fournir la valeur à afficher à la méthode refresh, il n'y a aucun calcul automatique intégré a l'objet graphique.

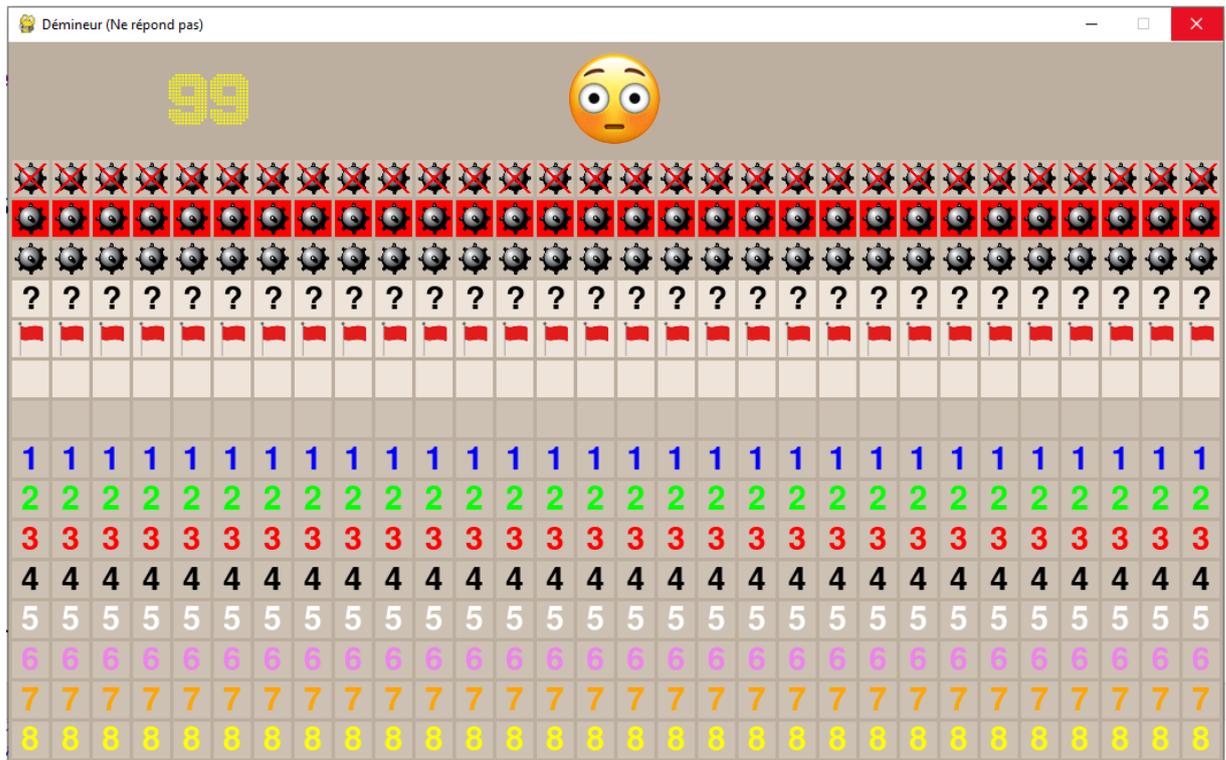
- **emo** est un paramètre optionnel qui correspond l'emoji à afficher au centre du bandeau supérieur de la fenêtre. Le chiffre (entier) correspond au numéro de l'image à afficher :



Images sous licence common creative : source pixabay.com

Par exemple :

```
grille = [[y for x in range(30)] for y in range(-6, 9)]  
GUI.refresh(grille, 99, 0)
```



- **gameOver() et gagne()** : Ces méthodes permettent respectivement d'afficher GAME OVER ou GAGNE !! en plein écran jaune sur fond rouge.

Par exemple :

```
GUI.gameOver()
```



- **waitClick()** : Cette méthode attend l'action d'un joueur. Elle gère trois types d'actions :
 - **demande fermeture de la fenêtre** : fermeture propre de la fenêtre pygame et fin du programme python.
 - **click sur la fenêtre** : retourne un tuple contenant les numéros (x, y, C) de la case choisie et une lettre C qui peut être :
 - 'D' ou 'G' qui correspond au bouton de la souris droit ou gauche qui a été cliqué
 - 'R' lorsque le click **gauche** (uniquement gauche) est relâché.
 - **click sur l'emoji** : retourne la lettre 'E'
 - **appui sur des touches spéciales** :
 - fleche RIGHT : retourne '_R';
 - fleche LEFT : retourne '_L';
 - fleche DOWN : retourne '_D';
 - fleche UP : retourne '_U';
 - touche BACKSPACE : retourne '_B';
 - touche RETURN : retourne '_E';
 - touche ESCAPE : retourne '_S';
 - **appui sur une autre touche du clavier** : retourne le caractère unicode correspondant.

Attention, une fois exécutée, on ne peut sortir de cette méthode que par l'une de ces quatre actions.

III- L'interface graphique version 2

Cette version a un fonctionnement identique à la précédente mais intègre en plus des fonctionnalités de chronométrage de la partie.

Vous devez importer la classe `GUIDemineur` depuis le module fourni :

```
from GUI_demineur.guiDemineur_V2 import GUIDemineur
```

- Les méthodes :
 - **resetTime ()** : Cette méthode permet de remettre le chronomètre à zéro.
 - **getTime ()** : Cette méthode renvoie le temps écoulé en seconde.
 - **stopTime ()** : Cette méthode stoppe le défilement du chronomètre.
 - **startTime ()** : Cette méthode autorise le défilement du chronomètre.
 - **chronolsEnable ()** : Cette méthode renvoie un booléen qui indique l'état du chronomètre. True s'il défile, False s'il est arrêté.

Pas d'image ici, il faut les tester en dynamique en visualisant les exemples inclus dans le module.