

# Interface graphique pour le projet de porte-avions

F. Barthélemy

25 avril 2014

## 1 Présentation de l'interface graphique

Pour le projet de porte-avions, nous vous proposons d'utiliser l'interface graphique disponible sur le site web du cours sous forme d'une archive zip qui comprend :

- Un fichier source java appelé `IGPA.java`
- Un dossier contenant des images appelé `images`
- Un fichier source exemple d'utilisation appelé `DemoIGPA.java`

Cette interface est donnée pour afficher le jeu. Il ne faut pas l'utiliser comme structure de donnée pour représenter le jeu. Par ailleurs, cette interface ne fait pas d'entrée d'information que ce soit par la souris ou le clavier. Les entrées se feront par le terminal.

**Il n'est pas nécessaire de comprendre le code de l'interface graphique pour l'utiliser. Il suffit de comprendre à quoi servent les méthodes de la classe principale pour réaliser le projet.**

Vous pouvez modifier l'interface graphique, mais pas pour faire le coeur du programme qui doit être séparé. Si vous la modifiez, ou si vous écrivez votre propre interface graphique, c'est toujours pour faire les entrées-sorties du programme uniquement.

## 2 Représentation du terrain dans l'interface

Le terrain est un damier de  $x*y$  cases. L'interface dispose de deux structures de données : une appelée `images` qui stocke des couples (entier, image). L'entier est une clé d'accès pour désigner une image. L'autre structure appelée `jeu` est un tableau d'entiers à deux dimensions qui représente les cases du terrain. Dans chaque case du tableau, il y a un entier qui est la clé d'accès à l'image à afficher à cette position. Par exemple, si dans `jeu[2][4]` contient 17, alors il faut afficher en colonne 2, ligne 4 l'image associée à la clé 17 dans la structure `images`.

## 3 Méthodes et étapes

La classe principale de l'interface graphique s'appelle `IGPA`.

Les composants publics de cette classes sont les suivants.

- Le constructeur `IGPA` prend deux paramètres : le nombre de colonnes et le nombre de lignes du terrain. Ce constructeur crée un objet java, mais rien ne s'affiche à ce moment-là. Il faut auparavant définir les images à utiliser et le contenu des différentes cases.
- `declarerImage` prend en paramètre un entier et un string. Le string doit être le nom d'un fichier de type image. Cette méthode ajoute un nouveau couple (entier,image) dans l'interface.

- `definirTerrain` prend un paramètre, tableau d'entier à deux dimensions. Il doit avoir les mêmes dimensions que le nombre de colonnes et de lignes définis dans le constructeur. Le contenu de ce tableau est recopié dans l'interface pour définir les images à afficher à chaque position.
- `modifierCase` prend trois entiers en paramètres : un position x,y et la clé de l'image à afficher à cette position.
- `creerFenetre` affiche à l'écran l'état courant du jeu. En principe, cette méthode ne doit être appelée qu'une fois dans le programme.
- `reafficher` sans paramètre refait l'affichage à jour. Les deux méthodes `definirTerrain` et `modifierCase` changent les valeurs des cases stockées dans l'interface, mais l'affichage n'est pas mis à jour par ces méthodes. Pour afficher des changements, il faut appeler une un plusieurs fois `definirTerrain` ou `modifierCase` puis terminer par un appel à `reafficher`.
- `fermer` sans paramètre ferme la fenêtre graphique. Cette méthode doit être appelée à la fin du programme pour le terminer proprement.

Le scénario type d'utilisation de l'interface est le suivant :

1. phase d'initialisation :
  - création de l'objet IGPA par appel du constructeur (`new IGPA(x,y)`).
  - déclaration de toutes les images utilisées : un appel à `declarerImage` pour chaque image.
  - définition de la valeur initiale du terrain par un appel à `definirTerrain` ou plusieurs appels à `modifierCase`.
  - création de la fenêtre d'affichage avec un appel à `creerFenetre`.
2. traitement en boucle au fil du déroulement du jeu :
  - modifier les valeurs des cases par un appel à `definirTerrain` ou plusieurs appels à `modifierCase`.
  - afficher les changements avec un appel à `reafficher`.
3. terminer l'exécution du programme : appeler la méthode `fermer`.

## 4 Le dossier des images

Les images sont stockées dans un dossier appelé `images`. Si vous compilez directement dans le terminal, ce dossier doit être dans le même dossier que vos fichiers source (par exemple `DemoIGPA.java`).

Si vous utilisez Eclipse ou Netbeans, il faut placer ce dossier au bon endroit dans votre projet. Il ne faut pas que ce soit avec les sources dans `src`, mais un cran au-dessus dans l'arborescence (le dossier `images` doit être un *frère* des dossiers `src` et `bin`).

Vous pouvez rajouter des images de votre choix dans ce dossier. L'interface est conçue pour utiliser des images de 48\*48 pixels. Elle ne fonctionnera pas bien avec des images d'une autre dimension.