**Compte-rendu - IHM - TP n°6**

GRAGLIA Valentin

TD3 – TP5

### A/ Découverte du premier programme wxWidgets

1. Il s’agit de gardes d’inclusions. Cela permet de ne compiler qu'une seule fois cette partie du code.

2. L'Appli0 contient les informations relatives à l'application qui va être lancée. Principale0 contient ce qui concerne la fenêtre principale de l'application.

3. Cette directive de préprocesseur permet d'inclure le fichier wx.h qui contient lui-même toutes les inclusions de toutes les bibliothèques wxWidget.

### Concernant Principale0

1. Principale0 hérite de la classe wxFrame. On le constate grâce au « : public wxFrame », ajouté avant sa définition.

WxFrame étant elle-même une classe pour créer des fenêtres déplaçables par l'utilisateur, cela semble logique.

2. a ┤ Le constructeur de Principale0 appelle lui-même le constructeur de wxFrame (on le voit dans le .cpp). La fenêtre Principale0 étant héritée, c'est justifiable.

b ┤ On a gardé un seul paramètre car cela nous l'est permis via le fait que les autres paramètres ont une valeur par défaut dans leur déclaration. Ici, ce qui nous intéresse est seulement de choisir le nom de notre fenêtre principale.

c ┤ On utilise ici le paramètre NULL car notre fenêtre ne possède aucune parentée. En effet, étant la fenêtre principale, c'est elle-même qui sera la mère des autres fenêtres s’il vient à y en avoir ; personne ne l'appelle.

3. Les éléments fils de Principale0 devront être crées dans la définition de Principale0. Nous, par la suite, utiliserons pour chaque élément un pointeur déclaré dans son header.

4. Pour créer une autre fenêtre on peut utiliser wxFrame dans Appli0.cpp par exemple.

### Concernant Appli0

1. Appli0 hérite de la classe wxApp. On le voit dans son header (« : public wxApp »).

2. Cette méthode initialise les infos et paramètres de l'appli (lors du lancer)

3. Cette macro permet de rendre visible dans les autres fichiers du programme une fonction wxGetApp qui permet d'obtenir une référence à notre application.

4. Appli0.h est le header de la classe et principale0.h est inclus car on l'utilise pour créer la fenêtre principale de l'appli.

5. Il n'y a pas de constructeur dans cette classe car on utilise les macros de wxWidgets.

6. Cette macro-ci permet de « dire » à wxWidgets comment créer une instance de la classe d'application.

7. Cette méthode charge la fenêtre puis la rend visible => Pour une visibilité totale une fois chargée. Enfin, elle lance la boucle des messages qui est le propre de la programmation évènementielle.

Plus généralement :

Il faudra modifier la boucle qui attend qu'un message soit envoyé et qui l'interprète.