

Examen OS - Windows API

Polytech ESR 4

On souhaite concevoir une application Windows avec des multiples fonctionnalités et une instance unique, qui présente une fenêtre de taille initiale 500x500 avec un menu permettant de quitter l'application, de changer la couleur du fond, de basculer en plein écran et de dessiner de manière persistante et non-persistante.

A. Concevoir une classe **ComplexWind** qui modélise une fenêtre normale (bordure déformable, titre) et qui demande à la construction le titre et l'adresse d'un **RECT** pour la position initiale. Elle a le titre "**Fenêtre complexe**" et elle affiche initialement un fond blanc et un menu composé des éléments suivants :

- **Application**
 - **Quitter**
 - **Quitter dans 3s**
 - **Quitter dans 10s**
- **Fenêtre**
 - **Fond blanc**
 - **Fond rouge**
 - **Fond vert**
 - **Fond bleu**
 - **Plein écran**

La fenêtre doit demander l'autorisation de fermeture à l'utilisateur si on la ferme à l'aide des boutons ou du menu système. Par contre, la fermeture par le menu utilisateur se fait automatiquement (sans confirmation) soit immédiatement, soit elle est retardée de 3s ou 10s. **Conseil :** utiliser des *timers* avec l'ID 1 et 2 avec les bonnes périodes pour détruire la fenêtre au bon moment.

B. Concevoir une méthode **ComplexWind ::SetBack** destinée à changer la couleur du fond de la fenêtre avec celle qui arrive en paramètre. Cette méthode récupère l'ancien *brush* (pinceau) utilisé pour dessiner le fond de la fenêtre (où se trouve-t-il ? comment l'obtenir ?), le remplace par un autre, nouvellement créé, qui a la bonne couleur, puis demande de redessiner la fenêtre (**InvalidateRect**) et enfin détruit l'ancien *brush*. Appeler cette méthode autant de fois que nécessaire quand l'utilisateur appuie sur les éléments de menu "**Fond *****" ou quand il appuie sur les touches "R" (red/rouge), "V" ou "G" (green/vert), "B" (bleu) ou "W" (white/blanc).

C. Faites afficher d'une manière persistante, votre nom et prénom dans le coin inférieur droit de la fenêtre (choisir la meilleure API pour l'affichage et/ou contrôler l'alignement du texte).

D. Faire en sorte que l'appui sur l'élément de menu "**Plein écran**" transforme la fenêtre dans une autre qui n'a plus de bordure, ni titre, ni menu et qui occupe entièrement l'écran. Pour cela effectuer dans l'ordre :

- Eliminer le menu de la fenêtre : le sauvegarder dans un attribut de la classe (initialement nul) à l'aide de **GetMenu** puis remplacer le *handle* de menu de la fenêtre avec un *handle* nul à l'aide de **SetMenu**.
- Changer le style de la fenêtre puis la repositionner, en s'inspirant de la méthode **SetStyle** (attention les paramètres ne sont pas forcément les mêmes). Pour avoir la taille de l'écran on peut utiliser **GetSystemMetrics** avec **SM_CXSCREEN** ou **SM_CYSCREEN**.

Faire en sorte que l'appui sur la touche **Echap (VK_ESCAPE)** fasse revenir le menu : utiliser le *handle* de menu précédemment sauvegardé dans un attribut de la classe.

E. On souhaite que le déplacement de la souris sur la fenêtre en appuyant bouton gauche provoque l'affichage non-persistant de petits rectangles jaunes à contour noir. Leur position sera définie par les 2 points successifs de la souris pendant son déplacement. La position précédente de la souris et sa validité sont gardés comme attributs puis réactualisés après chaque nouveau déplacement de la souris. Pour dessiner le rectangle courant utiliser le **minimum** et le **maximum+1** des coordonnées disponibles (courantes et précédentes). Prévoir un *brush* de couleur jaune comme attribut (attention à la gestion correcte des ressources : sélection, destruction...). Prévoir aussi que l'appui sur le bouton droit puisse provoquer le dessin non-persistant d'ellipses définies exactement de la même manière.

L'appui sur la touche **Suppr** (**VK_DELETE**) doit provoquer l'effacement intégral de la fenêtre.

Vous avez le choix d'utilisation entre GDI ou GDI+ mais GDI+ vous apporte un bonus.

F. Faire en sorte que l'application ne puisse exécuter qu'une seule instance à la fois, toute instance supplémentaire doit afficher une boîte de message pour avertir l'utilisateur et quitter aussitôt. Lire attentivement l'aide des fonctions **CreateXXX** qui explique, une fois le *handle* obtenu, comment utiliser **GetLastError** pour faire la différence entre un nouvellement créé et un qui existait déjà. Placer le code nécessaire dans la fonction **_tWinMain**.

L'ensemble de votre programme doit se compiler en version Unicode et Multibyte/Ansi ainsi qu'en version Debug et Release.