

UE: MU5SP05

Lab 8 : Création et utilisation des bibliothèques statique

Note :

Lorsqu'on compile, si une des bibliothèques indiquée est statique, le code du fichier objet est inclus dans l'exécutable. Utiliser une bibliothèque statique permet donc d'exécuter le programme sur une carte qui ne comporte pas la bibliothèque. Vous pouvez par exemple développer un exécutable qui utilise une bibliothèque commerciale que vous avez achetée et que vous n'avez pas le droit de diffuser. En compilant avec la version statique, vous incluez le code nécessaire, et votre application pourra être exécutée sur une carte qui ne comporte pas la bibliothèque.

Une compilation qui utilise des bibliothèques statiques produira un exécutable qui occupera plus d'espace disque. Lorsque les fonctions de la bibliothèque sont très utilisées on ne prendra pas la version statique de la bibliothèque pour compiler. En effet, si toutes les applications de votre ordinateur dupliquaient le code des fonctions d'entrées/sorties (vers la console ou un autre flux), vous perdriez beaucoup d'espace disque inutilement.

1. Nous souhaitons créer une bibliothèque statique liboutils.a contenant les 2 fonctions max et min développée dans la partie 7. Copier les fichiers sources de ces deux fonctions max.c, max.h, min.c et min.h.
2. Lancer la compilation des deux fichiers max.c et min.c sans faire d'édition de liens.
3. Avec la commande ar créer un archive nommée liboutils.a. Rajoutez dans cet archive le fichier objet des différentes fonctions.
Hint : utiliser la commande ar r nom de la bibliothèque « fichiers.o à insérer »
4. Vérifier avec l'option t la liste des fichiers objet dans l'archive créer liboutils.a.
5. Générer un index pour l'archive via la commande ranlib. A quoi sert t-il ?

6. Créez un fichier main.c permettant de créer et d'initialiser un tableau de 7 éléments avec les valeurs suivantes: 13 12 5 1 19 13 17 et appeler les fonctions max et min pour calculer le max et le min.
7. Créer deux répertoires include et lib dans votre répertoire /opt
8. Déplacer avec la commande mv les fichier max.h et min.h dans include et liboutils.a dans lib
9. **Supprimer tous les fichiers .o dans votre répertoire courant** et supprimer également les max.c et min.c Maintenant, compilez le fichier main pour créer l'exécutable final en faisant appel à la bibliothèque liboutils que vous avez créée précédemment.

#Hint : Commande de compilation en faisant appel à une bibliothèque
gcc main.c -L/\$path-library\$ -I/\$path-include-file\$ -l\$library-name\$ -o executable
10. Tester le bon fonctionnement de votre programme.
11. Nous souhaitons rajouter dans la bibliothèque liboutils.a une nouvelle fonction arithmétique var. Récupérer le fichier var.c. Compiler ce fichier. Rajouter cette fonction à la bibliothèque liboutils.
12. Vérifier la présence du fichier var.o dans la bibliothèque.
13. Utiliser la commande ranlib et nm pour mettre à jour la l'indexe et vérifier la présence des fichier .obj des fonctions.

14. Utiliser la fonction `var` dans `main.c` pour calculer la variance du tableau. Compiler. Tester et vérifier le bon fonctionnement.

15. Nous allons maintenant utiliser les fonctions `ceil` et `floor` de la bibliothèque `math` standard existante. Dans le `main`, utilisez ces deux fonctions pour arrondir le résultat de la variance que vous avez déjà calculé.

16. Compiler, Exécuter et vérifier les résultats.