TP 2 (2025-10-24) — Compilation (noté)

Rendu papier, ou bien email avant 20:00

Décrivez votre réflexion — rédigez pour être lisible par un humain

1 - Configuration du shell

- 1. Créez un alias lll pour afficher la liste des fichiers dans le dossier courant en liste détaillée, avec des tailles faciles à interpréter, et avec de la couleur si possible.
- 2. Rendre cet alias persistant au redémarrage du shell
- 3. Programmez une commande (soit compilée, soit interprétée avec shebang)

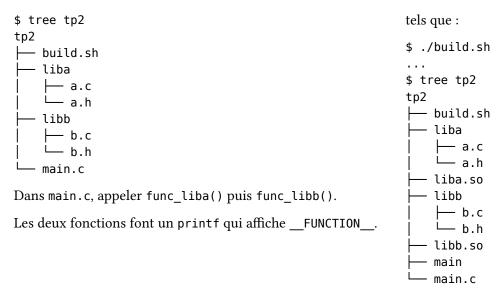
Par exemple dans ~/.local/bin ou ~/bin (pas déjà dans le \$PATH)

Par exemple un « cowsay », ou autre manipulation sur les argument, quelque chose de sympa

4. Rendre cette commande exécutable depuis n'importe où, de manière persistante

2 - Construire une librairie, avec un Makefile

1. Dans un dossier dédié, réaliser les fichiers suivants :



2. Remplacer build.sh par un Makefile

3 - Ordre de link

En dupliquant le dossier de l'exercice 2, dans main.c, appeler foo() uniquement.

Implémenter foo dans liba et libb; chacune affiche foo liba ou foo libb.

- 1. Est-il possible de link les deux bibliothèques au binaire ? L'ordre a-t-il un effet ?
- 2. Étudier l'utilisation de LD_DEBUG=symbols

4 - Show-bizz (bonus)

Écrire un programme qui affiche les nombres de un à cent :

- 1. sauf si c'est un multiple de 3, afficher « show »
- 2. sauf si l'écart avec le « show » précédent est un multiple de 4, afficher « bizz »
- 3. mais si 1. et 2. sont vrais, afficher « showbiz »
- $4.\,$ rendre configurables les valeurs 3/4 via une ou deux variables d'environnement

Pourquoi pas proposer plusieurs approches?

Comparer l'ordre de présentation du problème contre les structures de code choisies.