

TP Bases Outils pour l'Ingénieur

Exercice 1: Mise en place de l'environnement

1. Testez différents éditeurs et choisissez un éditeur en mode texte et un éditeur en mode graphique (si pas de préférence, texte: vim, graphique: vscode)
2. Modifiez le PS1 de votre terminal à votre goût
3. Créez un alias "ll" pour afficher la liste des fichiers dans le dossier courant en liste (-l) et avec des tailles lisibles (-h) (si supporté aussi avec de la couleur --color)
4. Persistez cet alias dans votre ~/.bashrc

Exercice 2: Faire un Makefile construire une Lib

Voici la hiérarchie de notre code:

```
.
├── build.sh (http://build.sh)
├── liba
│   ├── a.c
│   └── a.h
├── libb
│   ├── b.c
│   └── b.h
└── main.c
```

Pour l'instant nous n'avons que `build.sh` qui est un script shell appelant GCC.

1. Expliquez le fonctionnement de `build.sh` et les différents paramètres passés
2. Transposez ce fonctionnement dans un Makefile
3. Modifiez le Makefile pour construire les dossiers `liba` et `libb` comme des bibliothèques dynamiques.

Exercice 3: Faire un Morpion

1. Suivez le découpage du cours et créez les fichiers
2. Mettez en place le système de build Makefile
3. Implémentez le morpion (mode 2 joueurs)
4. Implémentez une IA

Exercice 4: Ordre de Link

Créez un main appelant une fonction foo située dans une bibliothèque liba ((qui affiche simplement foo_liba). Ensuite, créez une seconde fonction foo dans une bibliothèque libb (qui affiche simplement foo_libb).

1. Est-il possible de link les deux bibliothèques au binaire ?
2. L'ordre a-t-il un effet ?
3. Lancez avec la variables d'environnement LD_DEBUG=symbols