

Projet Système : Find.

I- Fonctionnalités implémentées :

Pour lancer le programme, nous avons fait un make qui se trouve dans src et qui génère un exécutable à la racine du projet.

Au cours de ce projet nous avons codé tous les prédicats (sauf le perm et exec), et ceux-ci correspondent au cahier des charges.

Notre programme accepte , 1 ou plusieurs chemins, suivis de prédicats (une alternance prédicat pur et impur est possible, mais changera l'affichage). Le programme gère les et implicites, ainsi que les OU (-o) et ET (-a) explicites, mais ne gère cependant pas le chainage, la négation ou les parenthèses pour marquer les priorités.

Il accepte les options -mindepth et -maxdepth, mais pas sort.

Si on ne possède pas la permission pour accéder aux stat d'un fichier, la fonction -ls affichera des points d'interrogation formatés au lieu de l'affichage habituel.

Les erreurs provoquent une fermeture propre du programme ainsi que la mise de la valeur de retour de ce dernier à EXIT_FAILURE (« 1 » dans la majorité des standards), tandis que des droits insuffisants sur un fichier provoquent seulement un warning sans changer la valeur de retour.

II- Améliorations :

La première amélioration est bien sûr de finir tout ce qui est demandé dans le sujet, comme la fonction perm, ou la fonction exec, mais aussi les opérateurs non implémentés ou le tri.

Mais après ça, quelques améliorations pourraient également être appliquées au programme :

- La mise en forme des paramètres dans un arbre, ce qui faciliterais beaucoup le traitement ultérieur dans process(), et cela permettrait aussi de jouer sur leur ordonnancement.
- la création d'une batterie de fonctions alternatives pour les prédicats, prenant en paramètre les stats d'un fichier plutôt que le path vers le fichier lui-même. S'il y a de nombreux prédicats, le gain pourrait être non négligeable.
- Enfin, au lieu d'utiliser un tableau de taille fixe et grande, il faudrait utiliser un tableau de taille plus réduite alloué dynamiquement au début de chaque appel récursif, avec possibilité de l'agrandir (realloc) au besoin.