

RAPPORT PROJET MINI_SHELL :
@author Mehdi WISSAD

Question 1 :

Le but est de créer la boucle de base de l'interpréteur avec des commandes simples c'est pour cela que j'ai utilisé une boucle infinie comme vue en cours ,TP et en utilisant « readcmd » déjà fournis .

Question 2 :

Pour cette question j'ai utilisé un print pour l'affichage , et la construction de ma session simple .

Question 3 :

Pour attendre la fin de la dernière commande lancée avant de passer à la lecture de la ligne suivante , j'ai utilisé le primitive wait()

Question 4 :

J'ai complété mon code en ajoutant deux commandes internes, qui sont exécuté directement par l'interpréteur sans lancer le processus fils j'ai utilisé un strmcp .

Question 5 :

Sur les taches de fonds , j'ai utilise les notions vu dans le tutoriel .

Question 6 , 7 :

j'ai créé deux fichiers un fichier .c et un autre .h afin de manipuler les processus .

```
/* Fonctions et procédures pour manipuler liste.*/  
/*Afficher la liste.*/  
/*la position du processus.*/  
/*IdPIDProc retourne l'IdPid du processus.*/  
/*Supprime les processus.*/  
/*Ajouter un processus .*/
```

Cela me facilitera la tâche après .

Pour mon handler j'ai utilisé un handler qui regroupe tous les signaux et selon le type de signale qui rentre j'ai choisi de faire un switch case et d'après le nom du signale j'effectue un traitement approprié .

J'ai trouvé une difficulté dans la création de ma liste de processus je n'arrive pas à l'afficher Ce qui me cause problème c'est que si cette fonction est juste je pourrais alors suivre les autres etapes et créer d'autre processus .

REMARQUE :

Dans mon archive j'ai deux minishell.c (contient la reponse de 1 à 5) minishell1.c (contient(6 et 7) en cours de realisation j'ai pas mal avancer dans ce qui est demandé mais il me faut plus de temps afin de mieux comprendre les notions et bien les appliquer)