

1. Objetivo

Implementar mini shell mediante el uso de llamadas al sistema: fork, exec, wait, exit.

2. Ejercicios de Laboratorio

2.1. Mini Shell

Implemente un pequeño mini-shell que reciba comandos ingresados por el usuario y los ejecute. Los comandos definidos serán los siguientes:

- ls <parámetro>: listar los archivos y carpetas ubicadas en so2018 de acuerdo al valor del parámetro utilizado en linux.
- cat <nombre_archivo>: mostrar el contenido del archivo pasado por parámetro.
- nano <nombre_archivo>: editar el contenido del archivo pasado por parámetro.
- gcc <nombre_archivo>: compilar el contenido del archivo pasado por parámetro.
- execute <nombre_archivo>: ejecutar el archivo compilado pasado por parámetro.
- exit -1: Cerrar shell

Utilice la siguiente guía de pasos para elaborar el programa:

```
main(void) {  
  
    printf(">");  
    /*Leer comando*/  
    while (/*Condicion de termino*/) {  
        if (fork() == 0) {  
            /*Ejecutar comando leído y recibir el estado. Imprimir mensaje de  
error en caso de no poder ejecutarse.*/  
        } else {  
            pid = wait(&estado);  
            printf(">");  
            /*Leer nuevo comando*/  
        }  
    }  
}
```

Pistas:

- Utilice la función strcmp() del paquete string.h para comparar el comando ingresado con algún string específico.
- Para los primeros 4 comandos utilice execlp() y para el quinto utilice execl().