

## Proyecto # 2 (10%)

### Comunicación entre Procesos

#### Pipes

#### Objetivos:

Los objetivos de este proyecto son:

1. Familiarizarse con la programación concurrente utilizando procesos.
2. Familiarizarse con mecanismos de comunicación entre procesos: pipes (nominales y no nominales) y señales.

#### Enunciado:

Implementar un programa llamado **UsoDisco**, que calcule de forma concurrente el espacio en disco utilizado por los archivos regulares a partir de un directorio usando el esquema maestro/esclavo.

Los programas:

```
- UsoDisco [-h] | [-n i] [-d directorio] [ salida ]
```

donde

- **-h** muestra por pantalla un mensaje de ayuda (sintaxis, descripción de parámetros, etc.) y termina.
- **-n i** nivel de concurrencia solicitado. Por defecto crea un solo proceso trabajador
- **-d directorio** especifica un directorio desde donde calcula el espacio utilizado. Por defecto hace el cálculo desde el directorio actual
- **salida** archivo que contendrá la lista de directorios y el espacio en bloques ocupado por los archivos regulares. Valor por defecto es la salida estándar.

Este es un ejemplo de una salida la ejecución desde el directorio Proyectos2000

```
240 ./Proyectos2000/Alg1
20  ./Proyectos2000/Orga/Interrup
120 ./Proyectos2000/Orga
12  ./Proyectos2000/Alg3/Conexo
44  ./Proyectos2000/Interf
88  ./Proyectos2000/DB
```

Recuerde la semántica que tiene la especificación de sintaxis de comandos:

- [ ]: Indica que lo que esté entre esos símbolos es opcional. Se debe especificar un valor por defecto.
- A | B : indica que se espera que el comando se puede ejecutar ya sea con A o con B, pero no los dos al mismo tiempo. A y B puede ser una opción o secuencia de opciones como en UsoDisco.
- -n i: indica que cuando se vea en la lista de parámetros -n el siguiente debe ser interpretado como i. Esto se usa para no fijar el orden en que se pasan los parámetros al programa.

En resumen, esto implica que las siguientes operaciones son invocaciones válidas:

- ./UsoDisco -h
- ./UsoDisco salida
- ./UsoDisco -n 4 salida
- ./UsoDisco -d dir salida
- ./UsoDisco -n 6 -d dir salida
- ./UsoDisco -d dir -n 8 salida
- ./UsoDisco
- ./UsoDisco -n 4
- ./UsoDisco -d dir
- ./UsoDisco -n 6 -d dir
- ./UsoDisco -d dir -n 8

Las siguientes son ejemplos de invocaciones no son válidas (puede haber más):

- ./usoDisco -h : el comando se llama UsoDisco
- ./UsoDisco -n salida : falta el nivel de concurrencia
- ./UsoDisco -d 4 salida : a menos que haya un directorio con nombre 4
- ./usoDisco -h salida : -h no puede ser usado con las otras opciones.

El programa debe crear un número (especificado con la opción ``-n i") de procesos (*trabajadores*), que realizarán el cálculo del espacio a partir del directorio pasado por la línea de comando.

El proceso principal (llamado *maestro*) funciona de la siguiente forma:

- Explora el directorio pasado en línea de comando; para cada archivo *directorio* encontrado agrega el nombre a la lista de archivos a ser explorados.
- De los procesos trabajadores recibe la sumatoria del tamaño en bloques de los archivos regulares encontrados en el directorio explorado por el proceso, el número de archivos regulares contabilizados y una lista con los nombres de directorios que encontró en esa revisión. Estos nombres serán agregados a una lista del proceso maestro que contiene los directorios por explorar.
- El proceso maestro le asigna un directorio, de la lista de archivos a explorar, a un proceso libre, elimina el nombre del directorio de la lista y marca a dicho proceso como ocupado.
- Cuando se terminan de explorar todos los directorios, se imprime la respuesta, y se terminan

todos los procesos trabajadores.

Los procesos trabajadores libres deben:

- Recibir del maestro el nombre del directorio a explorar
- Calcular el espacio usado en disco por los archivos regulares y producir la respuesta: a) total de bloques ocupado por los archivos regulares almacenados en el directorio procesado y b) la lista de directorios encontrados.
- Al terminar, pasar la respuesta al maestro y cambiar su estado a libre
- Esperar por un nuevo directorio a explorar, e iterar
- No termina hasta que reciba una orden explícita del proceso maestro para terminar su ejecución.

La salida debe indicar<sup>1</sup>: directorio examinado (camino completo) y la cantidad de bloques ocupado por los archivos regulares en dicho directorio.

Adicionalmente a la salida en el archivo, el programa deberá escribir por pantalla un reporte que indique: el número de procesos creados, número de directorios examinados y el número de archivos contabilizados.

**La comunicación entre el proceso maestro y los procesos trabajadores debe llevarse exclusivamente usando pipes y señales.**

## Observaciones importantes:

1. SI LOS PROGRAMAS NO SE PUEDEN COMPILAR O NO CORREN, NO SERÁN CORREGIDOS. Los programas deben poder compilarse y ejecutarse en cualquiera de las estaciones (Linux) del LDC. Si Ud. realizó el programa en su casa o en la USB, pero en alguna otra plataforma hardware-software, debe asegurarse antes de la entrega que su proyecto funcione en las estaciones Linux antes mencionadas.
2. Para la versión de procesos debe utilizar las llamadas *fork* y *exec*
3. No deben descuidar la utilización de la memoria.
4. Deben ajustarse a los formatos de salida, sin embargo pueden complementarlo con gráficos
5. Deben hacer uso del makefile, si no se entrega un archivo Makefile, se considerará que su proyecto no puede ser compilado.
6. Deben hacer un programa modular y claro. Recuerden usar librerías y headers (archivos .h).

## El proyecto deberá ser entregado:

**Lugar:** en salón de clases:

**Hora:** **1:30 pm** del 21 de Marzo (Lunes de la semana 11).

### Sobre la Entrega

La entrega del programa debe hacerse antes de la hora de clase del día Lunes 21 de Marzo. Ud deberá colocar el archivo con su proyecto en Aula Virtual, para lo cual deberá crear el directorio Proy2 dentro de la carpeta documentos de su grupo. Note que debe estar suscrito a algún grupo en aula Virtual

---

<sup>1</sup> Las líneas deben estar ordenadas por el nombre del directorio.

para poder optar a esta opción, ***no espere al día de la entrega para notificar que tiene problemas o que no se ha registrado***. En este directorio colocará el archivo tar.gz que contenga los fuentes de su programa. No se corregirán proyectos que no sean colocados de esta forma.

Adicionalmente, el mismo 21 de Marzo, Ud. debe entregar al comienzo de la hora de clase el código fuente impreso en un sobre manila identificado con los nombres de los integrantes del grupo y ***el número del grupo***. El código debe tener una cantidad adecuada de comentarios. Siga la guía de estilo de programación en lenguaje C publicada en la sección de documentos.

### **NOTAS IMPORTANTES:**

1. **Proyecto que no sea entregado en el lugar, fecha y hora prevista, NO será recibido.**
2. **Proyecto que no cumpla con algunas de las especificaciones establecidas en este enunciado corre el riesgo de no ser corregido.**