

Memoria PCTR LAB 1.2

```
root@ubuntu1: ~  
root@ubuntu1:~# ps -e  
  PID TTY          TIME CMD  
    1 ?           00:00:02 systemd  
    2 ?           00:00:00 kthreadd  
    3 ?           00:00:00 rcu_gp  
    4 ?           00:00:00 rcu_par_gp  
    6 ?           00:00:00 kworker/0:0H-kb  
    8 ?           00:00:00 mm_percpu_wq  
    9 ?           00:00:00 ksoftirqd/0  
   10 ?           00:00:00 rcu_sched  
   11 ?           00:00:00 migration/0  
   12 ?           00:00:00 idle_inject/0  
   14 ?           00:00:00 cpuhp/0  
   15 ?           00:00:00 kdevtmpfs  
   16 ?           00:00:00 netns  
   17 ?           00:00:00 rcu_tasks_kthre  
   18 ?           00:00:00 kauditd  
   19 ?           00:00:00 khungtaskd  
   20 ?           00:00:00 oom_reaper  
   21 ?           00:00:00 writeback  
   22 ?           00:00:00 kcompactd0
```

Índice:

● Manager.c	3
● Contador.c	4
● Definitions.h	4
● Procesador.c	4
● Lista.c	4
● Ejecución	5

● Manager.c

- Es un gestor de procesos que crea varios procesos para buscar patrones en un archivo. Los argumentos que recibe son un archivo con un texto y otro archivo con patrones que buscar.
- `procesar_argumentos()` controla que los argumentos sean correctos y abre el archivo de entrada para contar el número de líneas que tiene.
- Se instala un manejador para SIGINT (Ctrl + c), si `signal()` devuelve un error, se imprime un mensaje de error y se finaliza la ejecución.
- `manejador_senhal()` se encarga de finalizar procesos y liberar memoria.
- `procesar_patrones()` se encarga de guardar los patrones del archivo en una lista enlazada.
- Se crea una tabla con los procesos encargados de buscar patrones. Se determina el número de procesos que se necesitan con el número de líneas en el archivo de entrada más el número de patrones menos uno (se usa un índice cero).
- Para cada línea en el archivo, se lanza un proceso contador que cuenta el número de veces que aparece cada patrón en la línea, y luego se lanzan procesos procesadores para escribir el resultado en la salida estándar. Cada proceso se lanza en un proceso separado, y se almacena en la tabla de procesos.
- Una vez lanzados todos los procesos, se espera a su terminación y se liberan los recursos antes de terminar. Si el usuario interrumpe el programa, se llama al manejador de SIGINT para terminar los procesos y liberar recursos antes de finalizar.

● Contador.c

- Recibe una línea de texto y cuenta el número de palabras que posee la línea.
- La función detecta cuando se está dentro o fuera de una palabra y lo utiliza para ir incrementando un contador con el total de palabras.
- Al final el programa imprime el número de palabras que hay en una determinada línea.

● Definitions.h

- Define constantes y estructuras para gestionar una lista de procesos, además de funciones para crear, destruir, insertar y eliminar nodos en la lista.
- Se define una constante PATH_MAX para establecer el tamaño máximo de un path y evitar desbordamientos.

● Procesador.c

- Busca un patrón en un archivo de texto, recibe como argumento el nombre de un archivo y el patrón a buscar.
- Abre el archivo y lee cada línea buscando el patrón.
Si lo encuentra, imprime un mensaje indicando el número de línea y el PID del proceso que lo ha encontrado.

● Lista.c

- Implementa una lista enlazada.
- Tiene funciones para crear y destruir listas, inserción y eliminación de nodos en cualquier punto de la lista.

• Ejecución

- Bastará con abrir el proyecto “Practice_1-2” (MAKEFILE debe estar en la raíz de la carpeta que se abra) y ejecutar “make clean && make”

```

• gmnjpjn@LegionY540:~/ClonedRepos/Practices_PCTR_LAB/Practice_1-2$ make clean && make
rm -rf *~ core obj/ exec/ include/*~ src/*~
mkdir -p obj/ exec/
gcc -Iinclude/ -c -Wall -ggdb src/manager.c -o obj/manager.o
gcc -Iinclude/ -c -Wall -ggdb src/lista.c -o obj/lista.o
gcc -o exec/manager obj/manager.o obj/lista.o -lpthread -lrt
gcc -Iinclude/ -c -Wall -ggdb src/procesador.c -o obj/procesador.o
gcc -o exec/procesador obj/procesador.o -lpthread -lrt
gcc -Iinclude/ -c -Wall -ggdb src/contador.c -o obj/contador.o
gcc -o exec/contador obj/contador.o -lpthread -lrt
./exec/manager data/solution.txt data/patrones.txt
[MANAGER] 6 procesos creados.
[CONTADOR 5924] La linea '0' tiene 0 palabras
[CONTADOR 5926] La linea '2' tiene 0 palabras
[CONTADOR 5925] La linea '1' tiene 0 palabras
[PROCESADOR 5927] Patron 'clavito' encontrado en la linea 0
[PROCESADOR 5927] Patron 'clavito' encontrado en la linea 1
[PROCESADOR 5927] Patron 'clavito' encontrado en la linea 2
[PROCESADOR 5928] Patron 'calvito' encontrado en la linea 0
[PROCESADOR 5928] Patron 'calvito' encontrado en la linea 1
[MANAGER] Proceso CONTADOR terminado [5924]...
[MANAGER] Proceso CONTADOR terminado [5925]...
[MANAGER] Proceso CONTADOR terminado [5926]...
[MANAGER] Proceso PROCESADOR terminado [5927]...
[MANAGER] Proceso PROCESADOR terminado [5928]...
[MANAGER] Proceso PROCESADOR terminado [5929]...

[MANAGER] Terminacion del programa (todos los procesos terminados).

```