



| | | |
|---|---------------------------|---|
|  Universitat Autònoma de Barcelona | Sistemes Operatius |  |
| Curs acadèmic: 2015-2016 | | Codi assignatura: 102747 |
| Pràctiques : Processos i crides al sistema | | Centre: Escola d'Enginyeria |

OBJECTIU

- Conèixer els mecanismes de gestió de processos
- Conèixer la gestió de les comunicacions entre processos

MATERIAL

Per a la realització de la pràctica es treballarà amb el llenguatge **C** sota el sistema operatiu **Linux**.

A més, es disposa de la següent comanda d'ajut:

- Comanda **man** de **Linux**: `man -s 2 <funció de llenguatge C>`,
exemple: **man -s 2 write**

AVALUACIÓ

Aquesta pràctica es realitza durant dues sessions de pràctiques. A cada sessió de pràctiques s'avaluen quatre punts:

- A. Treball previ a la sessió realitzat fora del laboratori
- B. Treball durant la sessió de pràctiques
- C. Test d'avaluació de la pràctica, que es realitzarà al final de la segona sessió
- D. Informe final de la pràctica

La suma de la nota obtinguda en les dues sessions de pràctiques compon la nota final de la segona pràctica. En total, el pes d'aquesta pràctica suposa un terç de la nota final de les pràctiques de l'assignatura. L'alumne ha de tenir en compte que el treball previ a cada sessió serà avaluat durant la mateixa sessió. Per tant, es necessari que es realitzi de forma prèvia a les sessions de pràctiques. De la mateixa forma, el treball realitzar durant la sessió serà revisat i avaluat abans de que aquesta sessió s'acabi.

El test d'avaluació ha de ser aprovat amb una nota mínima d'APTE per poder tenir una nota final de la pràctica 2 de l'assignatura.

A. TREBALL PREVI

L'objectiu del treball previ es començar la sessió amb un programa esquemàtic dels objectius de la pràctica. A partir d'aquest esquema, s'anirà treballant per afegir les funcionalitats requerides per l'enunciat.

1. Creació de processos i canals de comunicació

L'esquema inicial de la pràctica consisteix en la implementació d'un programa que crea tres processos diferents: un procés màster, i dos processos "sensors". Cada procés sensor creat ha d'esperar un missatge del procés màster amb un número d'estació i un nom. Quan aquest darrer procés rep el missatge, mostra per pantalla el seu número i nom i finalitza la seva execució. Després de que cada procés sensor acabi, el procés màster també finalitza.

El codi que implementa aquest treball previ s'ha de compilar i executar correctament per a ser considerat com a vàlid i per continuar amb el treball durant la sessió de pràctiques consultant dubtes i demanant l'ajut necessari al professor de pràctiques.

B. DESCRIPCIÓ

L'aplicació a desenvolupar es un simulador d'un sistema que recull les dades d'un conjunt de sensors de temperatura distribuïts geogràficament. L'aplicació ha de llegir un fitxer d'entrada anomenat *estacions.txt*. Després, ha de crear una sèrie d'estacions de mesura o sensors on es simula la presa d'un conjunt de mesures de temperatura, es calcula el valor mitjà i es registra aquest valor promig per a cada estació en un fitxer de sortida *mesures.txt* on la darrera línia s'imprimeix el mínim de tots els valors de temperatura registrats per les estacions meteorològiques.

El fitxer d'entrada *estacions.txt* consta dels següents camps:

<NúmeroEstació> <NomEstació> <NumeroMesures>

Cada camp té els següents significats:

- NúmeroEstació: Identificador numèric de l'estació meteorològica
- NomEstació: Nom alfanumèric de l'estació
- NúmeroMesures: Número de mesures a prendre en l'estació identificada

El fitxer de sortida mesures.txt té la següent estructura:

<NúmeroEstació><NomEstació><temperaturaMitja>

On

- NúmeroEstació: Identificador numèric de l'estació meteorològica
- NomEstació: Nom de l'estació
- TemperaturaMitja: Valor de temperatura mitja mesurada a l'estació identificada

En aquest fitxer, la darrera línia es l'estació on s'ha produït la temperatura mitja més baixa.

L'aplicació està composta per diferents processos que implementen una funcionalitat diferenciada. Aquests processos són: màster, estació i anàlisi. L'esquema de relació entre els processos està definit en la figura.

El procés màster és l'encarregat de crear la resta de processos estació i anàlisi i estableix una estructura de comunicació on el màster es comunica amb els processos estacions i aquests envien les seves dades al procés anàlisi.

Amb més detall, les funcionalitats requerides per cadascun dels processos és la següent:

Màster

Inicialment, l'aplicació ha de crear un procés màster. Aquest realitza els següents passos:

- Creació dels processos estacions. S'han de crear 10 processos d'aquest tipus.
- Creació d'un procés anomenat anàlisi.
- Lectura línia a línia del fitxer estacions.txt
- Enviament de cada línia del fitxer llegit en el pas anterior a cada procés estació creat. Un procés estació només rebrà una única línia del fitxer estacions.txt
- Espera a que finalitzin tots els processos estacions i el procés anàlisi per poder finalitzar la seva pròpia execució

Estació

El procés estació realitza els següents passos:

- Esperar a que el procés màster l'hi envii el seu número, nom i el número de mesures a prendre.
- Simular el procediment de presa de valors de temperatura: el procés ha de generar N mesures de temperatura entre 0 i 50 graus on aquest valor N és el tercer paràmetre rebut des de'l procés màster. Després, el procés calcula el valor mig de temperatura de tot el conjunt de valors rebuts i dorm 10 segons .
- Un cop es desperta, envia al procés anàlisi el seu número, nom i valor promig calculat.
- Finalitza la seva execució

Anàlisi

El procés anàlisi ha de realitzar els següents passos:

- Esperar a que algun procés estació li envii les seves dades i mesures calculades.
- Després de rebre la informació respecte a una estació meteorològica, escriu al seu fitxer de sortida mesures.txt el valor rebut
- A partir del valor rebut de temperatura mitja, calcula el valor mínim rebut fins al moment
- Quan hagi rebut la informació de totes les estacions, escriu el valor mínim rebut amb l'identificador de l'estació i el seu nom
- Finalitza la seva pròpia execució.

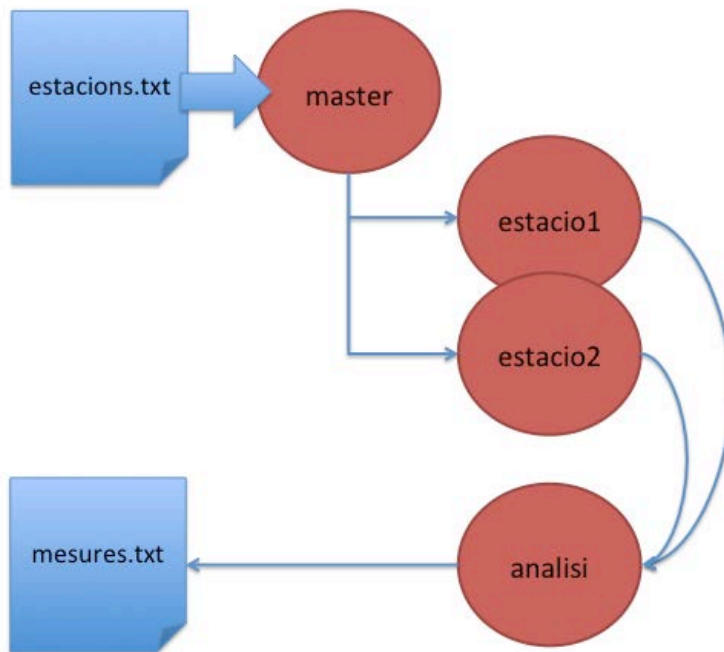


Figura: esquema de blocs dels processos i els fitxers d'entrada i sortida

TIPS & TRICKS

Revisar les transparències i explicacions de les classes de teoria corresponents a la secció Crides del Sistema.

Assegurar-se d'incloure les llibreries necessàries per al codi dissenyat.

Es valorarà com un aspecte molt important la implementació de mecanismes de control d'errors, especialment en el moment de realitzar crides al sistema.

Per a compilar el codi s'ha de fer servir el compilador "gcc". Es pot fer servir la següent comanda en el mateix directori on estigui ubicat el fitxer amb el codi font:

```
gcc -o nom_de_executable codi_font.c
```

Per conèixer més detalls sobre diferents opcions de compilació, feu servir el manual del gcc amb la comanda: `man gcc`