
Sistemes Operatius

Cosgrove system

Index

Contenido

Index.....	1
1. <i>Contingut</i>	2
<i>a. Descripció dels requisits</i>	2
<i>b. Disseny</i>	3
<i>i. Diagrames</i>	3
<i>ii. Com s'han assolit els requeriments</i>	3
<i>iii. Estructures de dades</i>	4
<i>iv. Estimació temporal</i>	5
2. <i>Conclusions</i>	6
3. <i>Bibliografia</i>	7

1. Contingut

a. Descripció dels requisits

En aquesta pràctica de l'assignatura de Sistemes Operatius es demana dissenyar un sistema de comunicació que permeti connectar els telescopis de l'Observatori Fabra, als quals s'han acoblat el procés McGruder que permet recollir les dades que capta l'aparell. Aquestes dades s'aniran guardant a diferents fitxers, i seran enviades a un procés central anomenat Lionel que es dedicarà a controlar tot el sistema. A part, s'hauran d'anar realitzant varis càlculs sobre aquestes dades que es van rebent, com que aquests càlculs poden ser bastant pesats, s'ha decidit que un altre procés anomenat Paquita s'encarregarà de realitzar aquests càlculs a mesura que es van actualitzant les dades rebudes a Lionel. Per a poder completar correctament aquesta pràctica s'han establert un seguit de fases que permetin als dissenyadors seguir una pauta concreta i avançar en la realització del sistema. Cada té uns requisits que s'han de complir i seran explicats a continuació.

- Fase 1:

En aquesta primera fase s'ha d'implementar el procés McGruder. Aquest serà inicialitzat amb un fitxer de configuració el qual indicarà uns paràmetres que McGruder tindrà en compte a l'hora de treballar. Aquests paràmetres són cada quant temps ha de revisar si ha arribat algun fitxer nou per part del telescopi, la IP i el port al qual s'haurà de connectar al procés Lionel. L'única tasca d'aquesta fase es consultar el directori pertinent i detectar si hi ha algun fitxer o imatge a enviar.

- Fase2:

En aquesta segona s'ha de configurar el Lionel de una forma similar als McGruder, llegint un fitxer de configuració que conte la IP a connectar-se, el port per on escoltar els McGruder, el port per comunicar-se amb els McTavish i un últim paràmetre que indica un temps en segons utilitzat per parametritzar els enviaments a la fase 3.

Apart, s'ha de implementar la comunicació entre McGruder i Lionel a través del port especificat i implementar la fase inicial del protocol de comunicació entre els dos.

- Fase 3:

En aquesta fase s'implementa l'enviament de dades. Aquestes dades poden ser o be fitxers de text o be imatges. Juntament amb el enviament de dades, s'ha d'implementar el tractament d'aquestes mitjançant els fitxers kalkun. Un cop completat el enviament de les imatges, aquestes han de ser reconstruïdes i mostrades tal i com eren en un principi, garantint la seva integritat mitjançant la comparació del md5sum de la imatge abans i després de ser enviada.

- Fase 4:

En aquesta última fase, es necessari implementar el procés Paquita, el qual esta encarregat de tractar les dades tan bon punt arribin al servidor. El tractament de les dades inclou el nombre de imatges rebudes, el seu pes total en KB, nombre de fitxer rebuts i la mitjana de constel·lacions de tots els arxius rebuts. A mes a mes, del últim fitxer de dades astronòmiques haurem de guardar el nombre de constel·lacions existents, mitjana de densitats i valor màxim i mínim de magnituds.

b. Disseny

i. Diagrames

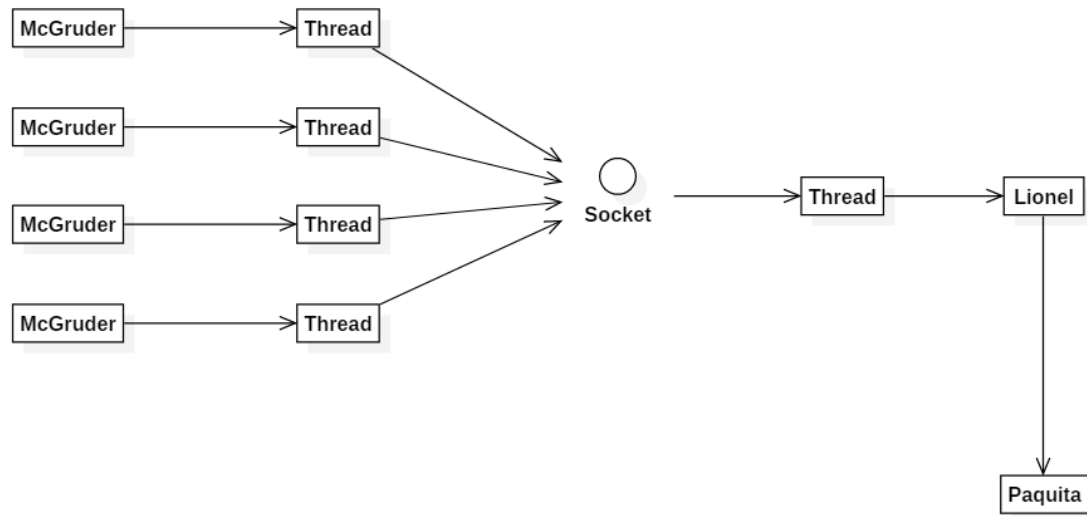


Figura 1 – Diagrama representatiu del sistema Cosgrove

Com es pot observar en la figura 1, el sistema compta amb una sèrie de clients McGruders. Aquests es comuniquen amb el servidor Lionel mitjançant un thread per no bloquejar el fil d'execució, i un socket per a poder transmetre dades. Un cop el servidor ha rebut les dades, el procés annex Paquita realitza el seu tractament.

ii. Com s'han assolit els requeriments

En base als coneixements de classe, s'han pres una sèrie de decisions per a la implementació dels diversos aspectes, especificats amb més detall a continuació.

- Comunicació Lionel – McGruder: Fan us de threads per evitar el bloqueig dels seus fils de execució i fan us de sockets per enviar-se la informació entre ells.
- md5sum: S'implementa mitjançant un fork ja que per a implementar-lo es necessari fer us de la funció `execl` la qual substitueix el codi actual per el que se li especifica a aquesta funció. Per últim, per retornar el valor obtingut amb l'execució del md5sum, es fa us de una pipe entre el fork fill i el fork pare.
- Procés Paquita: De forma similar al md5sum, aquest procés es duu a terme dins de un fork ja que també fa us de la funció `execl`. Per comunicar la informació entre Lionel i Paquita, s'ha fet us de una cua de missatges.
- Fitxers kalkun: Per a evitar problemes de concurrència entre els n McGruders a l'hora d'escriure sobre el fitxer kalkun (o la variable on es guarden les dades a escriure al fitxer) s'ha fet us de semàfors.
- Per assolir certs requeriments del enunciat, també ha sigut necessari re configurar certs signals del sistema com serien el `sigint`, `sigkill` i `sigalarm`.

iii. Estructures de dades

A continuació s'expliquen les diverses estructures de dades utilitzades:

```
typedef struct {  
    char* name;  
    int checkTime;  
    char* IP;  
    int port;  
} Telescope_info;
```

Telescope_info: Estructura on s'emmagatzema la informació llegida del fitxer de configuració dels McGruders.

```
typedef struct {  
    char* serverIP;  
    int mcGruderPort;  
    int mcTavishPort;  
    int time;  
} Lionel_config;
```

Lionel_config: Estructura on s'emmagatzema la informació llegida del fitxer de configuració del Lionel.

```
typedef struct {  
    char type;  
    char* header;  
    unsigned short int length;  
    char* data;  
} Data;
```

Data: Estructura on s'emmagatzema la informació llegida de la trama prèvia al enviament de informació per part dels McGruder.

```
typedef struct {  
    size_t size;  
} mem;
```

mem: Estructura utilitzada per a emmagatzemar la mida d'un fitxer

```
typedef struct {  
    int numConst;  
    float avgDensity;  
    float maxMagnitude;  
    float minMagnitude;  
} Last_Astronomical_Data;
```

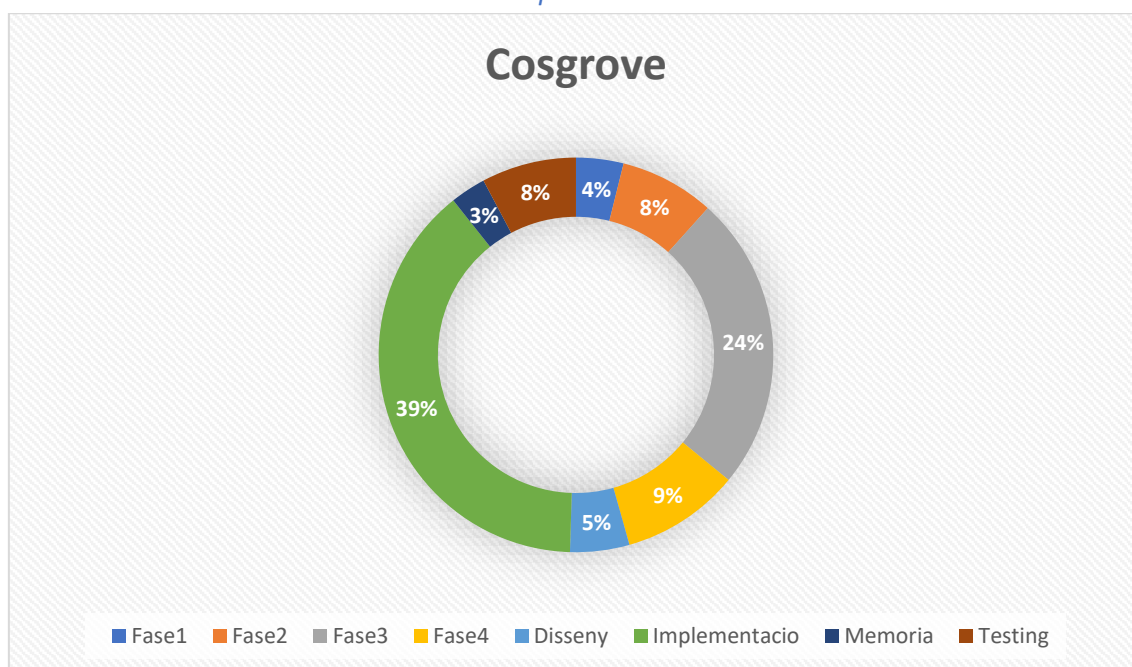
Last_Astronomical_Data: Estructura emprada per emmagatzemar la informació demanada del últim fitxer de dades astronòmiques rebut.

```
typedef struct {  
    char* extension;  
    size_t fileLength;  
    char* creationDate;  
} File_Info;
```

File_Info: Estructura emprada per emmagatzemar les dades pertinents a un fitxer, previ al seu enviament.

Apart, també s'han fet us de arrays dinàmics, implementat mitjançant mallocs i reallocs, per a realitzar tot tipus de tasques, però en especial per a guardar les dades dels enviaments de fitxers de text i d'imatges per a després escriure-les als fitxers kalkun.

iv. Estimació temporal



2. Conclusions

Aquesta practica ens demanava aplicar els coneixements teòrics vists a classe. Tot i que aquests coneixements son senzills d'entendre, el treballar en un entorn real com aquest ens ha permès observar que aplicar-los de forma efectiva pot resultar un xic complex.

De igual forma que la implementació de les diverses eines apreses durant el curs ha suposat un repte, també ho ha sigut el generar un bon disseny que permetés no tan sols realitzar el que demanava l'enunciat, si no que també ho fes de forma efectiva i eficaç. Ha sigut en aquest punt on s'ha presentat la major dificultat de la practica, al ser necessari l'esborrat sencer de certes funcionalitats dissenyades inicialment de forma molt poc optima.

Parlant també del disseny, la solució de la practica no es única, però hem optat per la solució presentada ja que creiem tant es una solució fàcil d'entendre i d'escalar així com el fet de que suposava un esforç menor de implementació en front de les demes propostes tingudes en consideració.

3. Bibliografía

Llibre Unix per Jordi Salvador Pont

https://estudy.salle.url.edu/pluginfile.php/651518/mod_folder/content/0/LlibreUNIX.pdf?forcedownload=1

Programacion de socket en C per CreativeCommons

http://www.chuidiang.org/clinux/sockets/sockets_simp.php

Ejecucion de procesos (execl) per ULPGC (Universidad de las Palmas de Gran Canaria)

http://sopa.dis.ulpgc.es/prog_c/PROCES.HTM

Forks in C per GessksforGeeks

<https://www.geeksforgeeks.org/fork-system-call/>

Mailboxes per MTU (Michigan Technological University)

<http://www.csl.mtu.edu/cs4411.ck/www/NOTES/threads/mbox.html>