

CONCEPTES AVANÇATS DE SISTEMES OPERATIUS (CASO)

Facultat d'Informàtica de Barcelona, Dept. d'Arquitectura de Computadors, curs 2014/2015 – 1Q

Pràctiques de laboratori

Mesures de rendiment

Material

La vostra instal·lació de Linux.

Mesures de rendiment

Per mesurar el rendiment d'aplicacions i sistemes operatius habitualment usem diverses mètriques:

- Temps d'execució
- Acceleració (speedup): la relació entre el temps d'execució en seqüencial i el temps d'execució en paral·lel
- Ample de banda (bandwidth): la relació entre les dades transmeses i el temps que s'ha invertit en transmetre-les
- Latència: Cost d'iniciar operacions o comunicacions

Les mesures preses poden tenir una certa variabilitat, per això és bo repetir els experiments un cert número de cops. Per exemple, fem l'experiment 10 cops i...

- Temps d'execució: fem la mitjana dels temps obtinguts en els 10 experiments
- Speedup: fem les mitjanes de les 10 executions seqüencials i de les 10 executions en paral·lel i l'speedup es calcula com a la seva relació
- Bandwidth: fem la mitjana dels temps d'execució dels 10 experiments i calculem el bandwidth o bé fem la mitjana dels diferents bandwidths obtinguts
- Latència: mitjana de les latències obtingudes en N experiments

També és bo calcular la desviació estàndard de les mesures, que ens donarà una idea de la seva variabilitat respecte la mitjana.

Eines de suport

L'eina més bàsica per a prendre mesures de temps en Linux/UNIX és **gettimeofday(...)**. És una crida a sistema que retorna el temps transcorregut des de l'1 de gener del 1970¹. Veure el seu manual (`$man gettimeofday`).

Amb aquesta crida podem obtenir el temps abans i després del codi que volem mesurar i fer la diferència.

Linux proporciona altres eines, en forma de comandes, que ens permeten analitzar l'execució del sistema o de les aplicacions: `top`, `ps`, `vmstat`, `iostat`...

¹ http://en.wikipedia.org/wiki/Unix_time

Exercicis

1. Escriviu un programa que calculi el temps que triga una crida a sistema senzilla. Per exemple, aquestes són interessants:
 - **sbrk(0)**, que només ha de retornar el *break point*, apuntant a la primera adreça invàlida després del *heap*. Veieu la diferència provant també amb **sbrk(inc)**, amb *inc* > 0. Quin "inc" podeu posar sense comprometre el sistema?
 - **sched_yield()**, que només suggereix al sistema operatiu que si troba un altre flux per executar, canviï de context.
 - **getpid()**, que retorna el pid del procés.
 - **fork()/waitpid()**, per crear i esperar un procés fill.

Per a fer-ho, feu programes que executin aquestes crides milions de vegades (desenes de milers en el cas del `fork()`) i preneu el temps. Podeu usar el programa *sleep_1.c*, que us mostra com prendre el temps de la crida a sistema *sleep*.

Feu una taula amb els temps d'execució que obteniu. Per què els temps d'execució són tan diferents?

Syscall	sbrk(0)	sbrk(inc)	sched_yield()	getpid()	fork/waitpid
Execution time (in microsecs.)					

2. Des d'un usuari no privilegiat (no root), executeu els dos programes que teniu cadascun de vosaltres de l'exercici de teoria (diguem-los `write-to-disk1` i `write-to-disk2`), sobre els dos discos que teniu a la màquina. Per exemple, podeu triar la partició de swap de sda (és `/dev/sda6` als PCs de les sales de la FIB), i un fitxer local a `sdb`:

```
$ write-to-disk1 /dev/sda6      $ write-to-disk2 /dev/sda6
```

```
$ write-to-disk1 file           $ write-to-disk2 file    # file representa un fitxer en el vostre disc.
```

Com podeu fer que un usuari normal pugui obrir el dispositiu `/dev/sda6` per lectura/escriptura?

(Ajuda: investigueu com usar la comanda `$ newgrp` per fer això)

Modifiqueu els programes per poder canviar la mida de les dades enviades al disc. Un cop executats, feu una gràfica amb els resultats que obteniu de *bandwidth*, comparant els dos discos i els dos programes. A l'eix de les X situeu el tamany de les dades transmeses i a l'eix de les Y, el *bandwidth*. Podeu usar OpenOffice Calc, el GNU plot, el *jgraph*...

3. Si executeu el programa com a administradors (root), obteniu alguna diferència en els resultats?
4. Escriviu i executeu una aplicació similar a la del punt 2, però que llegeixi del disc. Feu una gràfica similar a la del punt 2, que mostri ara el *bandwidth* obtingut per les lectures. Compareu els resultats.
5. Useu les eines del sistema **vmstat** i **iostat** per veure el *bandwidth* obtingut en lectura i escriptura de fitxers quan executeu els programes `write-to-disk` i `read-from-disk`, per separat i alhora. Per una mida concreta (p. ex. 200 Mb), coincideixen amb els vostres números?

L'entrega, a través del Racó, serà anunciada oportunament

Entregareu la taula de l'apartat 1 i els programes i les gràfiques obtingudes a l'apartat 2.