

CONCEPTES AVANÇATS DE SISTEMES OPERATIUS (CASO)

Facultat d'Informàtica de Barcelona, Dept. d'Arquitectura de Computadors, curs 2023/2024 – 2Q

Pràctiques de laboratori

Mesures de rendiment

Material

El vostre usuari a la màquina Mac OS X. Aquest quadrimestre farem les proves de rendiment en Mac OS X.

(Alternativament, o a més a més) La vostra instal·lació de **Linux (l'Ubuntu Server que teniu del primer dia)**. Podeu fer aquest laboratori a Linux directament, o bé, repetir les proves que feu a Mac OS X i comparar els resultats.

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 nombre 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 execucions seqüencials i de les 10 execucions 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*...

En el codi de suport d'aquest laboratori us proporcionem un exemple que mostra com aconseguir la data/hora actuals i un altre per comprovar quina és la fórmula correcta per calcular diferències dels

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

temps presos amb *gettimeofday*.

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.

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 microseconds.)					

2. Podeu comprovar d'alguna manera que els programes executen realment la crida a sistema? (I no aprofiten el resultat retornat per la crida anterior)
3. Fer un programa que escrigui al disc 500 MBytes, mesurant el temps que triga a fer-ho amb *gettimeofday* i que imprimeixi els temps invertit i el bandwidth que ha aconseguit la transferència d'informació

Modifiqueu els programa per poder canviar la mida de les dades enviades al disc. Un cop modificat i executat amb diverses mides, feu una gràfica amb els resultats que obteniu de bandwidth, comparant la mida de les dades. A l'eix de les X situeu la mida de les dades transmeses i a l'eix de les Y, el bandwidth. Podeu usar LibreOffice calc, el GNU plot, el jgraph...
4. <opcional> Si executeu el programa com a administradors (root), obteniu alguna diferència en els resultats?
5. <opcional> 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.
6. <opcional> 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. 500 Mb), coincideixen amb els vostres números?

L'entrega, a través del Racó, a data per confirmar

Entregareu la taula de l'apartat 1, la resposta a la pregunta 2, i els programes i les gràfiques obtingudes a l'apartat 3.