

Anàlisi del nombre de vols entre diferents aeroports

Octubre 2022

1 Introducció

Aquesta pràctica es pot considerar un preàmbul al que es farà a les següents pràctiques. En particular, en aquesta pràctica es proporciona un codi senzill que permet analitzar el nombre de vols que hi ha entre diferents aeroports dels Estats Units d'Amèrica.

La informació que es farà servir per fer aquest anàlisi és pública i està disponible a <http://stat-computing.org/dataexpo/2009/the-data.html>. En aquesta pàgina web hi ha disponibles fitxers de text pla tipus CSV que contenen informació, per a diversos anys, sobre els vols que s'han realitzat dintre d'aeroports dels Estats Units. A cada línia de fitxer s'inclou la informació d'un vol. Entre altra informació, s'inclou el codi del vol, el codi IATA de l'aeroport origen i destí, l'hora de sortida, l'hora d'arribada, el retard, etc.

En aquesta i les següents pràctiques s'utilitzaran els fitxers dels anys 2007 i 2008 per fer proves. Cadascun d'aquests fitxers és de tipus CSV i conté informació de més de 7 milions de vols. Per fer proves es proporciona un fitxer més petit que inclou només 10.000 vols.

L'objectiu d'aquesta pràctica és entendre el codi per tal de fer-lo concurrent, cosa que es farà a les següents pràctiques.

2 L'aplicació d'anàlisi de vols

L'aplicació necessita la següent informació com a entrada per fer l'anàlisi

1. Un fitxer que contingui els codis IATA dels aeroports a analitzar. En aquesta pràctica aquest fitxer s'anomena `aeroports.csv` i inclou 303 codis IATA d'aeroports dels Estats Units. Cada codi IATA té 3 lletres. S'ha fixat el nombre d'aeroports a 303 per fer més fàcil la programació de l'aplicació: aquest valor s'ha obtingut analitzant múltiples fitxers que contenen informació de milers de vols.
2. Un fitxer que contingui la informació dels vols entre els aeroports dels Estats Units. Aquesta informació s'ha obtingut de l'enllaç esmentat abans.

L'aplicació té per objectiu comptar el nombre de vols que hi ha entre els aeroports que s'han llegit abans fent servir la informació dels vols. Per això,

1. Es crea una matriu de mida 303×303 que emmagatzemarà el nombre de vols. La fila està associada a l'aeroport d'origen i la columna a l'aeroport destí.

2. Es llegeix el fitxer `aeroports.csv` i s'emmagatzemen els codis IATA en un vector. A la matriu esmentada fa un moment la primera fila està associada a l'aeroport de la primera fila del fitxer, la segona de la matriu està associada a la segona fila del fitxer, etc. És a dir, es fa un mapat senzill entre el fitxer `aeroports.csv` i la matriu.
3. Es procedeix a llegir el fitxer que conté la informació dels vols. Per a cada fila s'extreu la cadena que identifica el codi IATA de l'aeroport origen i destí. Per poder incrementar el valor corresponent a la matriu, es fa una cerca lineal de l'aeroport d'origen i destí al vector de codis IATA de l'aeroport d'origen i destí respectivament. Un cop s'ha trobat a quin index de la matriu es correspon l'aeroport d'origen i destí, s'incrementa el valor corresponent a la matriu esmentada al primer punt.
4. Finalment, un cop s'ha llegit tota la informació dels vols, s'imprimeix per pantalla, per a cada aeroport amb una o més destinacions, el nombre de destinacions que té.

L'algorisme no ha estat programat per ser eficient sinó que s'ha volgut mantenir la senzillesa per poder ser entès amb facilitat.

Per executar l'aplicació executeu

```
$ ./analisi aeroports.csv fitxer_petit.csv
```

3 Anàlisi del codi

Es demana fer les següents tasques. Per fer les proves feu servir els fitxers `2007.csv` o `2008.csv`.

1. Executar el codi i provar el seu funcionament. Observar que l'aplicació imprimeix per pantalla 303 aeroports d'origen. Identifiqueu aeroports coneguts? Nova York? Atlanta? Comenteu els resultats.
2. Quins són els aeroports amb més destinacions? Com us ho feu per saber-ho? Hi ha alguna comanda que us ho permeti saber?
3. Quan triga el codi executar-se sobre els fitxers `2007.csv` o `2008.csv`? Quina comanda es pot fer servir per mesurar el temps (real, usuari i sistema) d'un procés? Aquests fitxers es poden baixar des del campus virtual de l'assignatura.
4. Quants vols emmagatzemen aquests dos fitxers (`2007.csv` i `2008.csv`)? Amb quina comanda podeu esbrinar aquesta informació?
5. Callgrind és una eina que registra el nombre d'instruccions CPU que fan falta per executar cada funció així com l'historial de crides entre les funcions d'un programa. Callgrind no mesura el temps real d'execució, sinó que només mesura el temps de CPU. Aquesta eina és doncs útil per mesurar el temps de CPU de cada funció. Callgrind no inclou, però, el temps d'entrada-sortida associat al procés i, per tant, no és útil per analitzar un procés que tingui moltes operacions d'entrada-sortida.

Per defecte, les dades recollides consisteixen en el temps d'execució CPU a cada funció així com el mapat amb el codi font i un graf que relaciona les crides de les funcions entre sí incloent el nombre de vegades que cada funció ha sigut cridada. Un dels beneficis que té callgrind

respecte altres eines com gprof és el fet que el primer analitza també les crides realitzades a funcions de llibreries compartides (com, per exemple, la llibreria d'usuari que inclou funcions com *printf*, ...).

Per analitzar el codi fent servir el callgrind cal compilar el codi amb l'opció de debugger. Un cop compilat el codi executeu la següent instrucció

```
valgrind --tool=callgrind ./analisi aeroports.csv fitxer_petit.csv
```

Podeu provar també amb els fitxers 2007.csv o 2008.csv. El codi pot trigar una estona en executar-se (alguns minuts). Aquesta instrucció genera un fitxer anomenat **callgrind.out.<pid>**, on <pid> és l'identificador del procés executat, i que inclou informació de l'execució. Un cop ha acabat l'execució podeu fer l'anàlisi dels resultats de l'historial de crides executant

kcachegrind

Aquesta aplicació mostrarà per pantalla un munt d'informació respecte el temps d'execució de cada funció així com un graf associat a les crides realitzades. Es recomana esborrar el fitxer **callgrind.out.<pid>** cada cop que vulgueu tornar a executar el callgrind atès que sembla que la comanda kcachegrind llegeix tots els fitxers en cas que n'hi hagi múltiples d'aquest tipus.

1. 6. El codi entregat no és net, sinó que té errors de programació. Els podeu identificar? **Heu de trobar aquests errors, comentarlos a l'informe, i tornar el codi net amb el fitxer ZIP que heu de lliurar al campus virtual.** Una forma ràpida d'analitzar el codi és fer servir el valgrind, l'eina que va començar a fer servir a Sistemes Operatius 1, per analitzar problemes de programació. Recordeu que heu de compilar el codi amb una opció particular per fer servir el valgrind! Executeu el valgrind d'aquesta forma:

```
$ valgrind ./analisi aeroports.csv fitxer_petit.csv
```

Valgrind imprimeix els errors per la sortida estàndard d'error. Perquè no se us n'escapi cap redirigiu-la cap a un fitxer, així:

```
$ valgrind ./analisi aeroports.csv fitxer_petit.csv 2> errors.txt
```

4 Entrega de la pràctica

Es demana **entregar un fitxer ZIP** que inclogui un informe i el codi **analisi.c** arreglat. El nom del fitxer ha d'indicar el número del grup, seguit del nom i cognom dels integrants del grup (e.g. **P3_Grup07_nom1_cognom1_nom2_cognom2.zip**). L'informe a entregar ha d'estar en **format PDF** o equivalent (no s'admeten formats com odt, docx, ...). Indiqueu les respostes a les preguntes de la Secció 3. **No lliureu els fitxers 2007.csv o 2008.csv.**

L'informe estarà estructurat en una introducció, treball realitzat i conclusions. A la part d'introducció es descriu breument el problema solucionat (i.e. resum del que proposa en aquesta pràctica). A la part de del treball realitzat es responen les preguntes realitzades a la pràctica i es descriuen les proves que s'han realitzat. Per respondre les preguntes podeu seguir un fil conductor que us permeti descriure el treball realitzat. També podeu respondre de forma explícita a cadascuna de les preguntes (en aquest cas indiqueu clarament a quina pregunta esteu respondent en cada moment). Sigueu breus i clars en els comentaris i experiments realitzats, no cal que us

esteneu en el text. Finalment, a la part de conclusions es descriuen unes conclusions tècniques de les proves realitzades. Per acabar, es poden incloure conclusions personals.

Inclogueu, preferentment en format text, la comanda que heu executat i algun comentari breu descrivint el resultat si ho creieu necessari. En cas que preferiu incloure captures de pantalla en comptes d'incloure el resultat en format text, assegureu-vos que el text de la captura es pot llegir bé (és a dir, que tingui una mida similar a la resta del text del document) i que totes les captures siguin uniformes (és a dir, que totes les captures tinguin la mateixa mida de text).

El document ha de tenir una **llargada màxima de 5 pàgines** (sense incloure la portada). El document s'avaluarà amb els següents pesos: proves realitzades i comentaris associats, un 60%; escriptura sense faltes d'ortografia i/o expressió, un 20%; paginació del document feta de forma neta i uniforme, un 20%.