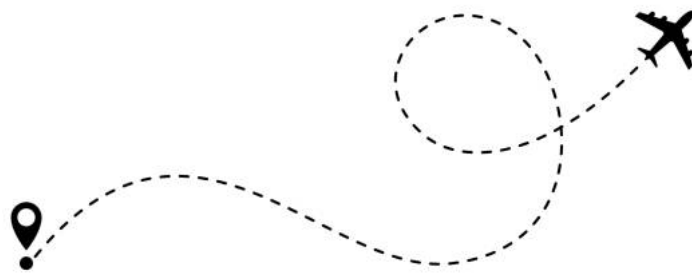

SISTEMES OPERATIUS 2:

Anàlisi del nombre de vols entre diferents aeroports



Lin Zhipeng
Oscar de Caralt

Curs 2022-2023

Introducció: Aquesta pràctica és una introducció perquè ens anem familiaritzant amb el codi que utilitzarem a les següents pràctiques (on veurem més d'aprop Callgrind i Valgrind) a través de l'anàlisi de dades de diferents aeroports dels Estats Units. L'objectiu d'aquesta pràctica és entendre el codi per tal de fer-lo concurrent, cosa que es farà a les següents pràctiques.

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.

El codi mostra per la pantalla 303 aeroports d'origen amb la quantitat de destinacions diferents corresponents.

```
Origin: ABE -- Number of different destinations: 0
Origin: ABI -- Number of different destinations: 0
Origin: ABQ -- Number of different destinations: 22
Origin: ABY -- Number of different destinations: 0
Origin: ACK -- Number of different destinations: 0
```

Amb els outputs podem veure el codi JFK que representa Aeroports per Nova York i el codi ATL que representa aeroports per Atlanta.

2- Quins són els aeroports amb més destinacions? Com us ho feu per saber-ho? Hi ha alguna comanda que us ho permeti saber?

Amb fitxer_petit.csv veiem que l'aeroport amb més destinacions és LAS, això es pot veure fent servir la comanda:

```
./analisi aeroports.csv fitxer_petit.csv | awk '$8>max {max=$8; row=$0} END {print row}'
```

```
oslab:~/Desktop/P3-Anàlisi de codi/codi> ./analisi aeroports.csv fitxer_petit.csv | awk '$8>max {max=$8; row=$0} END {print row}'
Origin: LAS -- Number of different destinations: 54
```

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.

El codi triga de 12 a 14 segons per a processar el fitxer. 2007.csv.

```
Tiempo para procesar el fichero: 12.259374 segundos
```

El codi triga de 10 a 11 segons per a processar el fitxer 2008.csv.

```
Tiempo para procesar el fichero: 10.933166 segundos
```

Amb la comanda `time` de Linux hem pogut obtenir els temps.

```
oslab:~/Desktop/P3-Anàlisi de codi/codi> time ./analisi aeroportos.csv 2007.csv
```

```
real    0m13.642s
user    0m9.584s
sys     0m3.227s
```

```
oslab:~/Desktop/P3-Anàlisi de codi/codi> time ./analisi aeroportos.csv 2008.csv
```

```
real    0m9.921s
user    0m6.546s
sys     0m2.841s
```

4- Quants vols emmagatzemen aquests dos fitxers (2007.csv i 2008.csv)? Amb quina comanda podeu esbrinar aquesta informació?

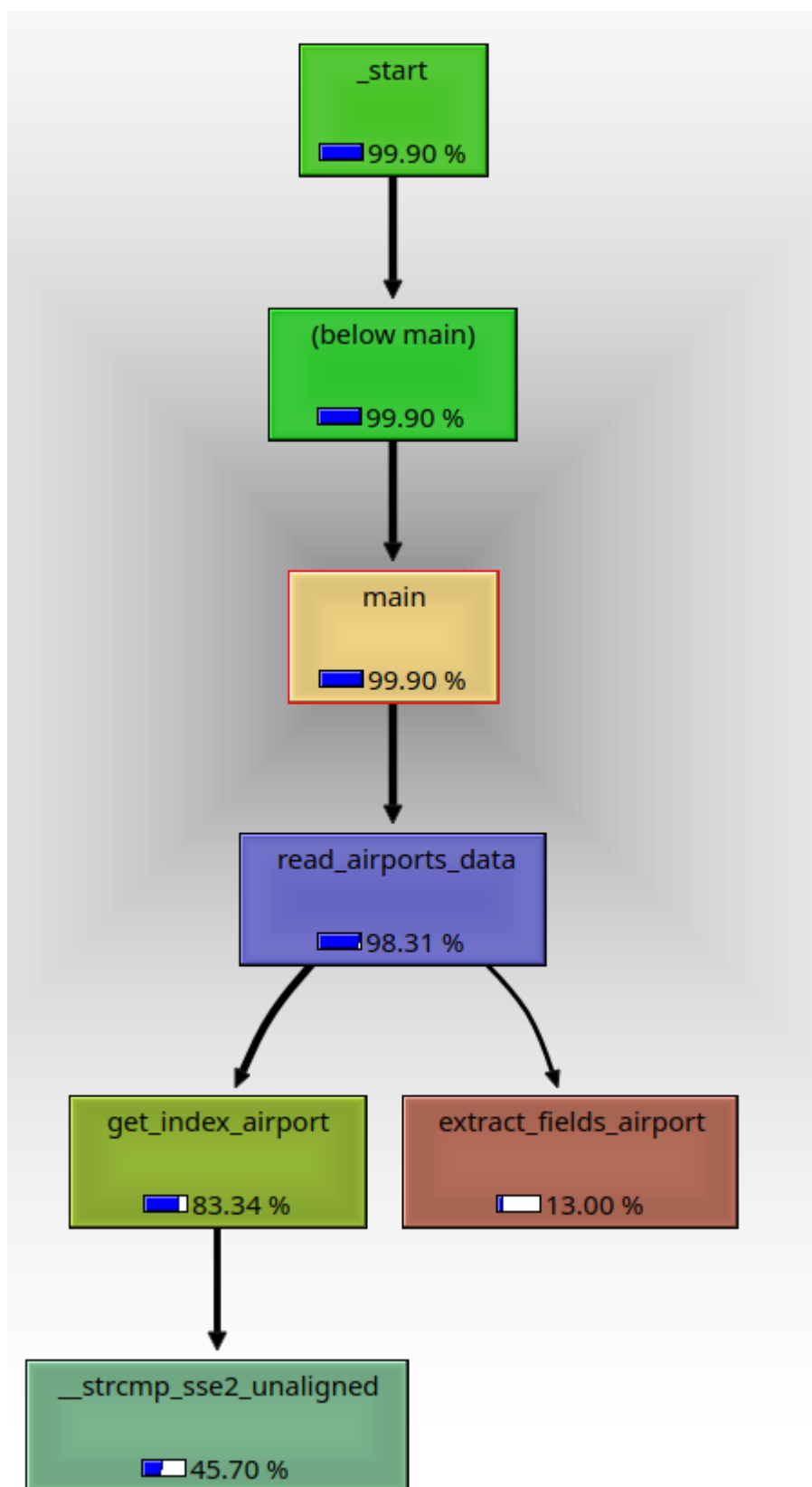
Podem fer servir la comanda `wc -l` per a consultar quantes línies de vols estan emmagatzemades a dins del fitxer.

```
oslab:~/Desktop/P3-Anàlisi de codi/codi> wc -l 2007.csv
7453216 2007.csv
oslab:~/Desktop/P3-Anàlisi de codi/codi> wc -l 2008.csv
7009729 2008.csv
```

5- Callgrind

```
Tiempo para procesar el fichero: 0.662115 segundos
==4111==
==4111== Events      : Ir
==4111== Collected : 142635951
==4111==
==4111== I   refs:      142,635,951
```

Incl.	Self	Called	Function	Location	Event Type	Incl.	Self	Short	Formula
100.00	0.00	(0)	0x0000000000001210	ld-2.31.so	Instruction Fetch	99.90	0.00		Ir
99.91	0.00	1	_start	analisi: start.S	Cycle Estimation	99.90	0.00	CEst = Ir	
99.91	0.00	1	(below main)	libc-2.31.so					
99.90	0.00	1	main	analisi					
98.31	0.32	1	read_airports_data	analisi					
83.34	37.64	20 000	get_index_airport	analisi					
45.91	45.91	3 363 147	__strcmp_sse2_unaligned	libc-2.31.so					
13.00	12.47	10 000	extract_fields_airport	analisi					
1.68	0.51	10 305	fgets	libc-2.31.so					
1.32	1.03	1	print_num_flights_sum...	analisi					
1.17	0.01	10 305	_IO_getline	libc-2.31.so					
1.15	0.48	10 305	_IO_getline_info	libc-2.31.so					
0.46	0.46	10 532	__memchr_sse2	libc-2.31.so					
0.33	0.33	20 303	__strcpy_sse2_unaligned	libc-2.31.so					
0.29	0.01	304	printf	libc-2.31.so					



6- El codi entregat no és net, sinó que té errors de programació. Els podeu identificar?

Executem la comanda `valgrind ./analisi aeroportos.csv fitxer_petit.csv` que mostrarà per la pantalla el següent missatge:

```
==4553== Memcheck, a memory error detector
==4553== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==4553== Using Valgrind-3.16.1 and LibVEX; rerun with -h for copyright info
==4553== Command: ./analisi aeroportos.csv fitxer_petit.csv
==4553==
==4553== Conditional jump or move depends on uninitialised value(s)
==4553==    at 0x400A3F: print_num_flights_summary (analisi.c:122)
==4553==    by 0x400E30: main (analisi.c:326)
```

L'error correspon la funció `print_num_flights_summary` i és degut a que no inicialitzem amb un valor la variable `num_flights`. Per a solucionar-ho hauríem d'inicialitzar la variable amb un valor.

```
void print_num_flights_summary(int **num_flights, char **airports)
{
    int i, j, num;

    for(i = 0; i < NUM_AIRPORTS; i++)
    {
        num = 0;

        for(j = 0; j < NUM_AIRPORTS; j++)
        {
            if (num_flights[i][j] > 0)
                num++;
        }

        printf("Origin: %s -- Number of different destinations: %d\n", airports[i],
num);
    }
}
```

Correcció:

```
// Reserva espacio para las matrices
airports = (char **) malloc_matrix(NUM_AIRPORTS, STR_CODE_AIRPORT, sizeof(char));
num_flights = (int **) malloc_matrix(NUM_AIRPORTS, NUM_AIRPORTS, sizeof(int));

//Init num_flights
for (int i = 0; i < NUM_AIRPORTS; i++){
    for (int j = 0; j < NUM_AIRPORTS; j++){
        num_flights[i][j] = 0;
    }
}
```

```
_petit.csv
==4738== Memcheck, a memory error detector
==4738== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==4738== Using Valgrind-3.16.1 and LibVEX; rerun with -h for copyright info
==4738== Command: ./analisi aeroports.csv fitxer_petit.csv
==4738==
Origin: AFE -- Number of different destinations: 0
```

Conclusions:

Amb aquesta pràctica ens hem pogut familiaritzar amb el codi i hem pogut recordar conceptes i algunes comandes que ens seran útils per a poder realitzar d'una millor manera. les properes pràctiques