Summary 3rd PL exam

15/12/2021

Session 3.1

Block in memory has same size as memory in cache.
Number of memory blocks in the main memory: words in the main memory / words per block.
Size of the cache memory: blocks in the cache x words per block x bytes/word.
LRU ⟶ Write-back.

How many accesses are performed in the execution?: View ⟶ Current Trace

If the only level in the memory system of this computer were the main memory, how many clock cycles would all the memory accesses consume?: Number of accesses x time per clock cycle in main memory.

Number of clock cycles consumed in all the memory accesses: N.misses x (Block size (B) / consecutive memory positions accessed in each memory access) x cycles memory transfer + N.hits x cycles hit.
Time: cycles x (1/freq.)

Cache hit rate if the memory access pattern is fully random: blocks in the cache / number of blocks in the main memory. (4 / 4096) x 100 in the example.
Average access time with fully random pattern: Acache x tcache + (1 - Acache) x tmem x N. of accesses
Required time to complete all the memory accesses: Average access time x N.accesses (80 × 125)

Session 3.2

Size of main memory: Memory blocks x memory block size.
Size of the cache memory: Memory addressable size *number of blocks (lines in direct mapped)* memory block size (words per block).

Replaced addresses: Address that has been replaced, not the new one.
Update addresses: If dirty bit = 1 and the address is replaced, then update. Note address replaced.

Session 3.3

Run with higher priority so it is not interrupted:

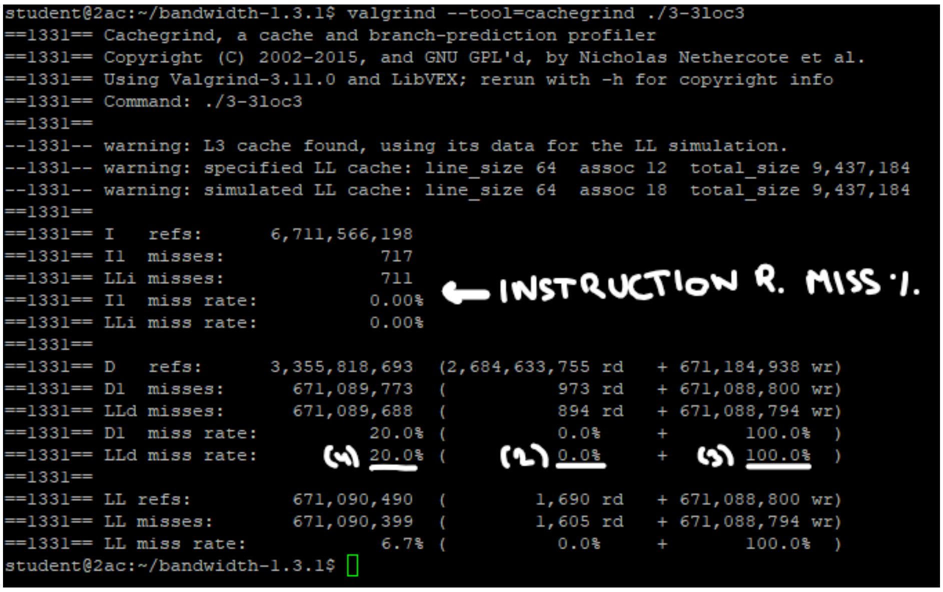
```
sudo nice -n -2 ./bandwidth32 --fastest
```

Hit rate en accesos a una matriz: Con bloques de 64 palabras, la primera palabra es miss, las siguientes son hit, por lo que el hit rate es 63/64×100.

Valgrind: Simulate the memory trace of the program.

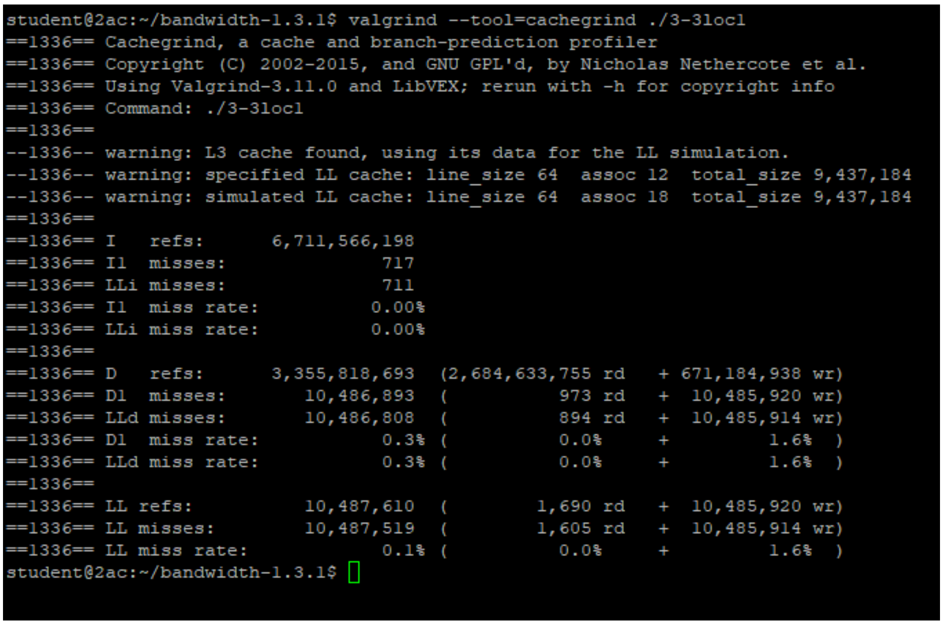
```
valgrind --tool=cachegrind ./3-3loc1
# If permission denied: chmod +x 3-3loc1
```

Table:



```
student@2ac:~/bandwidth-1.3.1$ valgrind --tool=cachegrind ./3-3loc3
==1331== Cachegrind, a cache and branch-prediction profiler
==1331== Copyright (C) 2002-2015, and GNU GPL'd, by Nicholas Nethercote et al.
==1331== Using Valgrind-3.11.0 and LibVEX; rerun with -h for copyright info
==1331== Command: ./3-3loc3
==1331==
--1331-- warning: L3 cache found, using its data for the LL simulation.
--1331-- warning: specified LL cache: line_size 64  assoc 12  total_size 9,437,184
--1331-- warning: simulated LL cache: line_size 64  assoc 18  total_size 9,437,184
==1331==
==1331== I   refs:      6,711,566,198
==1331== I1  misses:              717
==1331== LLi misses:              711
==1331== I1  miss rate:          0.00%    ← INSTRUCTION R. MISS %.
==1331== LLi miss rate:          0.00%
==1331==
==1331== D   refs:      3,355,818,693  (2,684,633,755 rd   + 671,184,938 wr)
==1331== D1  misses:      671,089,773  (          973 rd   + 671,088,800 wr)
==1331== LLd misses:      671,089,688  (          894 rd   + 671,088,794 wr)
==1331== D1  miss rate:          20.0% (          0.0%     +       100.0%  )
==1331== LLd miss rate:     (4)  20.0% (     (2)  0.0%     +   (3) 100.0%  )
==1331==
==1331== LL refs:         671,090,490  (        1,690 rd   + 671,088,800 wr)
==1331== LL misses:       671,090,399  (        1,605 rd   + 671,088,794 wr)
==1331== LL miss rate:            6.7% (          0.0%     +       100.0%  )
student@2ac:~/bandwidth-1.3.1$ 
```

(2): L3 Data Read Miss Rate.
(3): L3 Data Write Miss Rate.
(4): L3 Data Aggregated Miss Rate.



```
student@2ac:~/bandwidth-1.3.1$ valgrind --tool=cachegrind ./3-3loc1
==1336== Cachegrind, a cache and branch-prediction profiler
==1336== Copyright (C) 2002-2015, and GNU GPL'd, by Nicholas Nethercote et al.
==1336== Using Valgrind-3.11.0 and LibVEX; rerun with -h for copyright info
==1336== Command: ./3-3loc1
==1336==
--1336-- warning: L3 cache found, using its data for the LL simulation.
--1336-- warning: specified LL cache: line_size 64  assoc 12  total_size 9,437,184
--1336-- warning: simulated LL cache: line_size 64  assoc 18  total_size 9,437,184
==1336==
==1336== I   refs:      6,711,566,198
==1336== I1  misses:              717
==1336== LLi misses:              711
==1336== I1  miss rate:          0.00%
==1336== LLi miss rate:          0.00%
==1336==
==1336== D   refs:      3,355,818,693  (2,684,633,755 rd   + 671,184,938 wr)
==1336== D1  misses:       10,486,893  (          973 rd   +  10,485,920 wr)
==1336== LLd misses:       10,486,808  (          894 rd   +  10,485,914 wr)
==1336== D1  miss rate:           0.3% (          0.0%     +         1.6%  )
==1336== LLd miss rate:           0.3% (          0.0%     +         1.6%  )
==1336==
==1336== LL refs:          10,487,610  (        1,690 rd   +  10,485,920 wr)
==1336== LL misses:        10,487,519  (        1,605 rd   +  10,485,914 wr)
==1336== LL miss rate:            0.1% (          0.0%     +         1.6%  )
student@2ac:~/bandwidth-1.3.1$ 
```

Nivel	Tipo (I/D)	% fallos lectura	% fallos escritura	% fallos agregado
L1	instrucciones	0%	N/A	0%
L3	datos	0%	1,6%	0.3%

Session 3.4

See maps file with a pid:

```
more /proc/pid/maps
# replacing pid with the process identifier.
```

Compile using static libraries:

```
gcc -static 3-4maps.c -o 3-4maps-2
```

Tamaño de archivos en K:

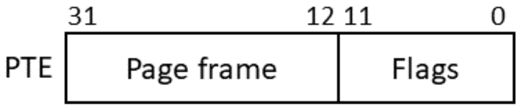
```
ls -l --block-size=K *maps-*
```

Linmem:

```
systemctl status linmem

cd /usr/src/atc/linmem
sudo make clean
sudo make install
```

Analysis of the address space of a task:



```
gcc 3-4print-vm.c 3-4print-pte.c -o 3-4print-vm -lmem
# Use -lmem flag.
```

Dynamic memory allocation

mmap:

```
gcc -Wall 3-4proclinux-2.c 3-4print-pte.c -o 3-4proclinux-2 -lrt -lmem
```

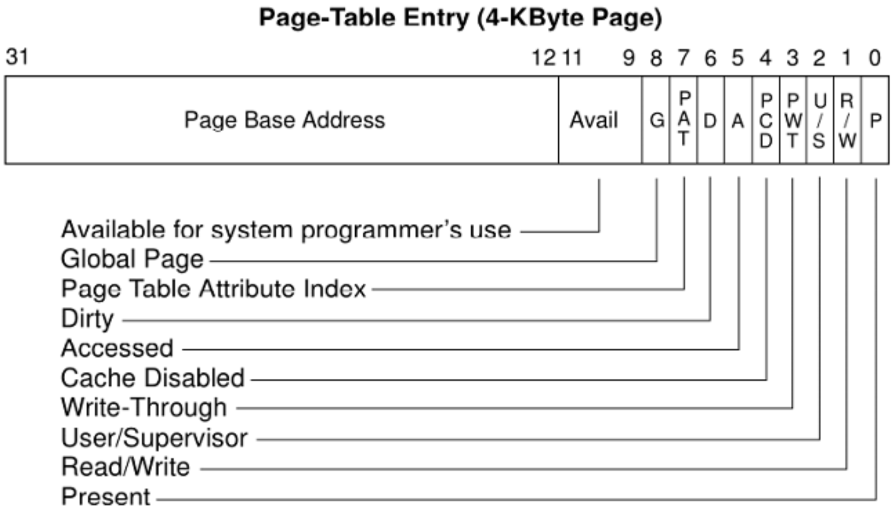
Lazy allocate: Si no se usa el espacio reservado, en realidad no se reserva.

munmap:

```
munmap(p, 16*1024);
```

"In what memory area of the task is the resquested memory located?": Check maps file in the virtual address area.

The page table:



Physical address assigned to system components:

```
sudo cat /proc/iomem
```