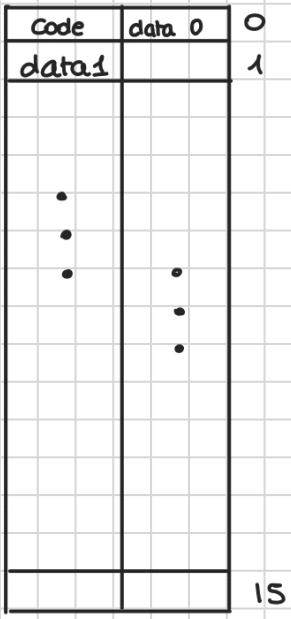
Exercise: 2-ways associative cache

block size 64 bytes
2-ways associative cache (16 sets)

```
.data
    x: .word
    n: .word 1024
.text
    |
ciclo: lw   t2, 0(t0)      1 + 1
       add  t3, t3, t2     3
       addi t0, t0, 4      4
       addi t1, t1, -1     5
       bne  t1, zero, ciclo  6
```



REMARK: when you have to replace a block, you discard the least recently used (LRU) block.
We need bits to do this.

To know how many instructions a block contains: → $\frac{\text{block size}}{4} = \frac{64}{4} = 16$

(4: one instr. 4 words)

Iteration 0: 1 miss, 6 memory accesses |

Iteration 1: 0 miss, 6 memory access |

⋮ // same

Iteration 16: 1 miss, 6 memory access

Iteration 17: 0 miss, 6 memory access

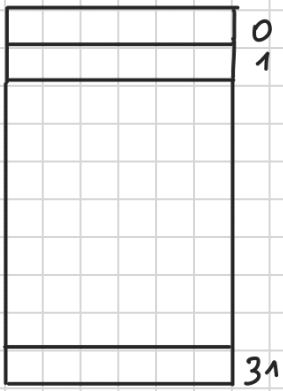
$$\text{MISS RATE (of 1st block)} = \frac{1 \text{ (n° misses)}}{16 \cdot 6} = 1\% \quad (1.01)$$

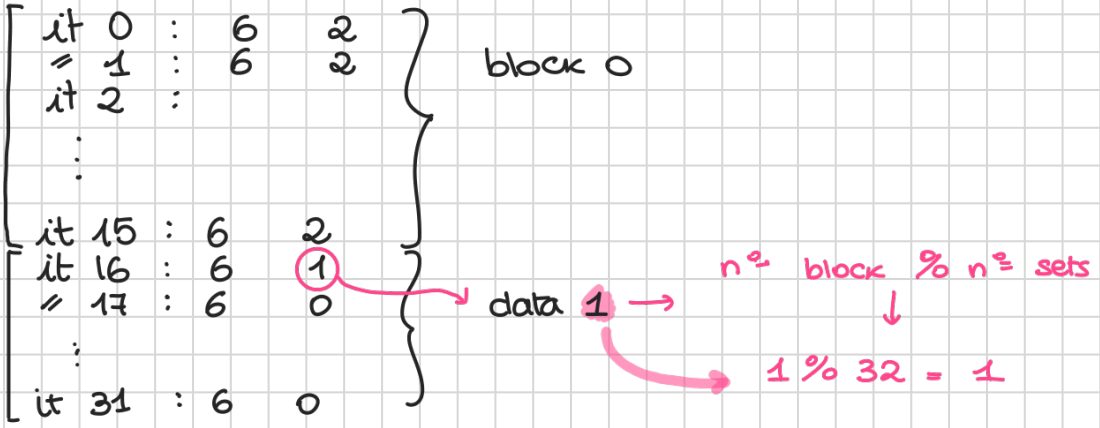
16: iterations; 6: memory access

Exercise: 1-way associative cache

block size 64 bytes
1-way associative cache (32 sets)

```
.text
        |
ciclo:  lw   t2, 0(t0)
        add  t3, t3, t2
        addi t0, t0, 4
        addi t1, t1, -1
        bne  t1, zero, ciclo
```





$$\text{MISS RATE} = \frac{32 + 31(1)}{16 \cdot 6 + 31(16 \cdot 6)} \quad (\text{or } 32(16 \cdot 6))$$

- 32: n° of misses from iteration 0 to iteration 15
- 31(1): n° of misses from iteration 16 to iteration 1023
- 32: n° of sets

Exercise: hits and misses

block size 16 bytes
2-ways associative cache (2 sets)

~~2~~ 0	4
~~1~~ 5	~~3~~ 1

0
1

M M M M M H M M H H

28, 48, 32, 92, 68, 52, 4, 24, 64, 24

LUR (least used recently).

28 M 28 : 16 = 1
1 mod. 2 = 1

48 M 48 : 16 = 3
3 mod. 2 = 1

32 M 32 : 16 = 2
2 mod. 2 = 0

92 M 92 : 16 = 5
5 mod. 2 = 1

68 M 68 : 16 = 4
4 mod. 2 = 0

52 H 52 : 16 = 3
3 mod. 2 = 1

4 M 4 : 16 = 0
0 mod. 2 = 0

24 M 24 : 16 = 1
1 mod. 2 = 1

64 H 64 : 16 = 4
4 mod. 2 = 0

24 H 24 : 16 = 1
1 mod. 2 = 1

Exercise: 1-way associative cache

block size 64 bytes
1-way associative cache (32 sets)

write back

miss rate?

```
.data
x: .word ____
y: .word ____
z: .word ____
n: .word 4.096
```

How many instructions in a block?

64 : 4 = 16 instructions in a block.
↑ block size ↖ word size (4 bytes).

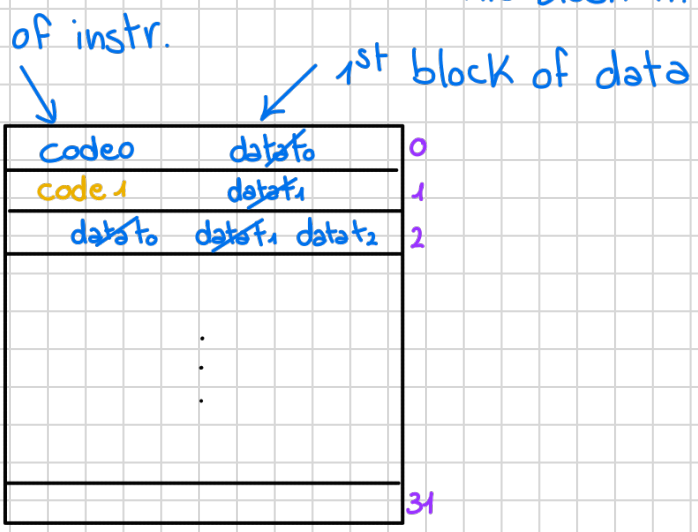
```
        .text                    M
pseudo  la   t0, x               H
instr.  la   t1, y               HH
        la   t2, z               HH
        lw   t3, n               HH

          H    M
ciclo:  lw   t4, 0(t0)
   H    lw   t5, 0(t1)           M
   H    add  t6, t4, t5
   H    sw   t6, 0(t2)           M
   M    addi t0, t0, 4
   H    addi t1, t1, 4
   H    addi t2, t2, 4
   H    addi t3, t3, -1          16 instr.
   M    bne  t3, zero, ciclo
```

to divide by 64, convert this addr. in binary and shift it 6 positions right. (2^6)

0x00400000 : 64 = block n. 000100000000...0

set n. $\frac{\text{block n.}}{32\ (2^5)}$ = last 5 bits of the block n.



BNE is out of the 1st block.

	M	H
i_0	8	12
i_1	6	12
i_2	6	12
⋮		
i_{15}	6	12
i_{16}	4	12
i_{17}	4	12
⋮		
i_{31}	4	12
i_{32}	3	12
i_{33}	3	12

TOT. miss rate $\approx \frac{8 + 15 \cdot 6 + 4 \cdot 16 + 3 \cdot 16 \cdot 30}{12 \cdot 16 + 12 \cdot 16 + 12 \cdot 16 \cdot 30} \approx 26\%$ (0,26)

misses 1° blocco; misses 2° blocco; misses 'till the end

12 accessi × 16 volte (1° blocco)

MISS RATE (of 1st block) $= \frac{8 + 15 \cdot 6}{16 \cdot 12} = 51\%$ (0,51)

n° misses; iterations; memory access