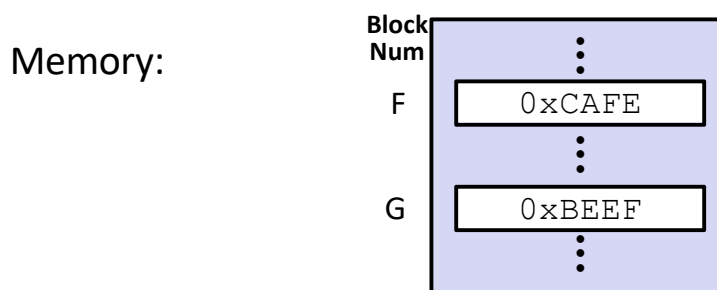
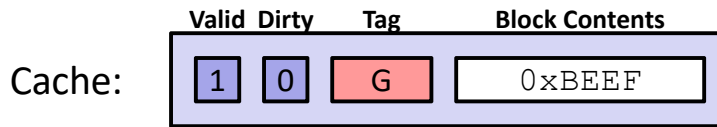


# Write-back, Write Allocate Example

1) `mov $0xFACE, (F)`    2) `mov $0xFEEF, (F)`    3) `mov (G), %ax`



41

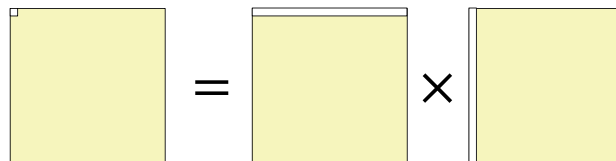
## Cache Miss Analysis Comparison

Ignoring  
matrix C

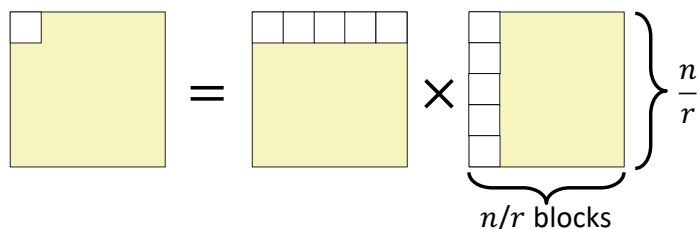
### ❖ Scenario Parameters:

- Square matrix ( $n \times n$ ) of doubles
- Cache block size  $K = 64 \text{ B} = 8 \text{ doubles}$
- Cache size  $C \ll n$  and three blocks ( $r \times r$ ) fit into cache:  $3r^2 < C$

### ❖ Naïve:



### ❖ Blocked:



42

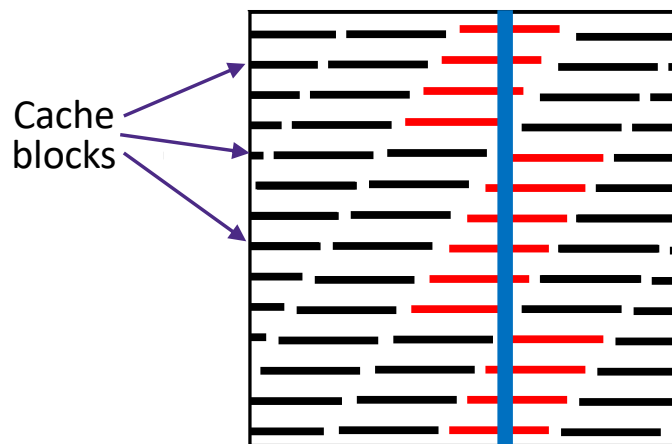
## Polling Question

- ❖ Which of the following cache statements is FALSE?
  - Vote at: <https://PollEv.com/wolfson>
  - A. We can reduce compulsory misses by decreasing our block size
  - B. We can reduce conflict misses by increasing associativity
  - C. A write-back cache will save time for code with good temporal locality on writes
  - D. A write-through cache will always match data with the memory hierarchy level below it
  - E. We're lost...

43

## Matrices in Memory

- ❖ How do cache blocks fit into this scheme?
  - Row major matrix in memory:



COLUMN of matrix (blue) is spread  
among cache blocks shown in red

44