

Recitation 11

Computer Architecture (section 1)

Cache Eviction Policies

Random

First in, first out (FIFO)

Least frequently used (LFU)

Least recently used (LRU)

Random Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

Random Cache Eviction

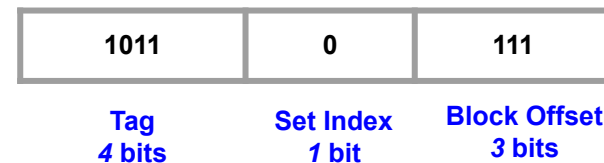
	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

Pick a line at random

Random Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	○○○○○○○○
	1	0001	○○○○○○○○
	1	1001	○○○○○○○○
	1	1111	○○○○○○○○
Set 1	0		
	0		
	0		
	0		

Read Request

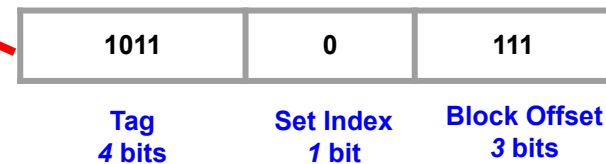


Pick a line at random

Random Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1011	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

Read Request



Pick a line at random

FIFO Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

FIFO Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

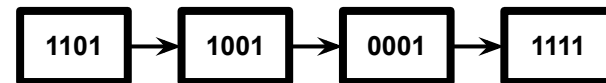
Use a queue (FIFO)

FIFO Cache Eviction

Set 0

Valid?	Tag	Cache Block (8 bytes)
1	1101	oooooooo
1	0001	oooooooo
1	1001	oooooooo
1	1111	oooooooo
0		
0		
0		
0		

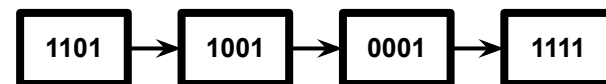
Set 1



Use a queue (FIFO)

FIFO Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	00000000
	1	0001	00000000
	1	1001	00000000
	1	1111	00000000
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

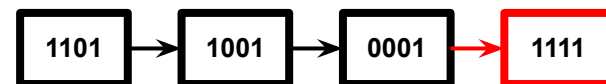
Set Index
1 bit

Block Offset
3 bits

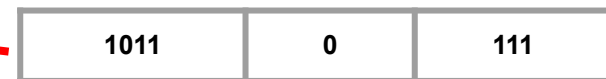
Use a queue (FIFO)

FIFO Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	00000000
	1	0001	00000000
	1	1001	00000000
	1	1111	00000000
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

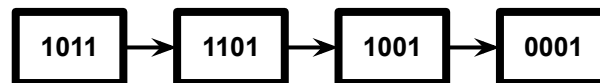
Set Index
1 bit

Block Offset
3 bits

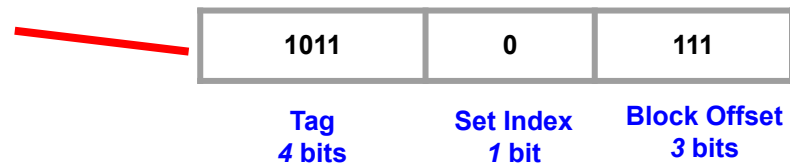
Use a queue (FIFO)

FIFO Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1011	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



Use a queue (FIFO)

LFU Cache Eviction

Set 0

Valid?	Tag	Cache Block (8 bytes)	Counter
1	1101	oooooooo	23
1	0001	oooooooo	42
1	1001	oooooooo	33
1	1111	oooooooo	12
0			
0			
0			
0			

Set 1

LFU Cache Eviction

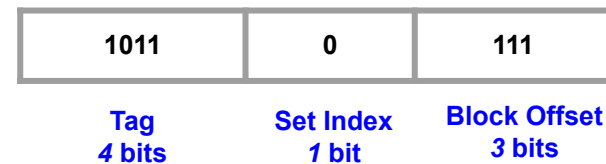
	Valid?	Tag	Cache Block (8 bytes)	Counter
Set 0	1	1101	oooooooo	23
	1	0001	oooooooo	42
	1	1001	oooooooo	33
	1	1111	oooooooo	12
Set 1	0			
	0			
	0			
	0			

Use access counters

LFU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)	Counter
Set 0	1	1101	oooooooo	23
	1	0001	oooooooo	42
	1	1001	oooooooo	33
	1	1111	oooooooo	12
Set 1	0			
	0			
	0			
	0			

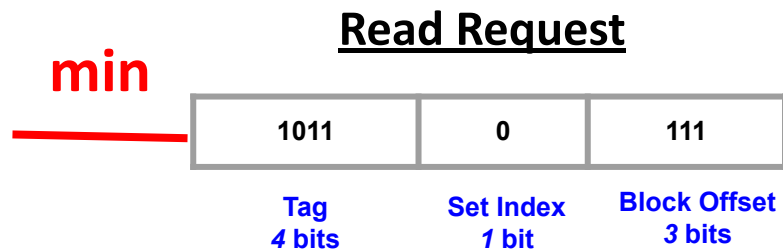
Read Request



Use access counters

LFU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)	Counter
Set 0	1	1101	oooooooo	23
	1	0001	oooooooo	42
	1	1001	oooooooo	33
	1	1111	oooooooo	12
Set 1	0			
	0			
	0			
	0			

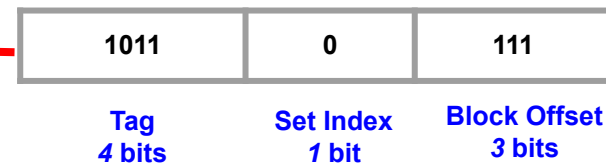


Use access counters

LFU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)	Counter
Set 0	1	1101	oooooooo	23
	1	0001	oooooooo	42
	1	1001	oooooooo	33
	1	1011	oooooooo	1
Set 1	0			
	0			
	0			
	0			

Read Request



Use access counters

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		

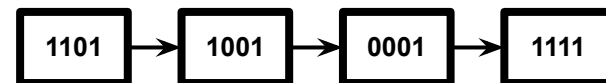
Use a stack (LIFO) +
hashmap

LRU Cache Eviction

Set 0

Valid?	Tag	Cache Block (8 bytes)
1	1101	oooooooo
1	0001	oooooooo
1	1001	oooooooo
1	1111	oooooooo
0		
0		
0		
0		

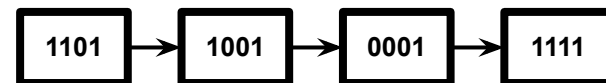
Set 1



Use a stack (LIFO) +
hashmap

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

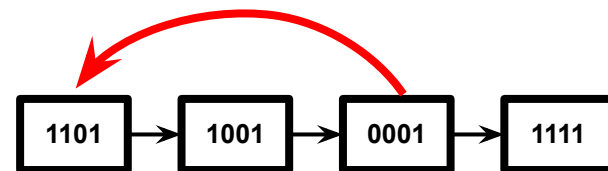
Set Index
1 bit

Block Offset
3 bits

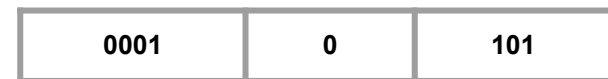
**Use a stack (LIFO) +
hashmap**

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

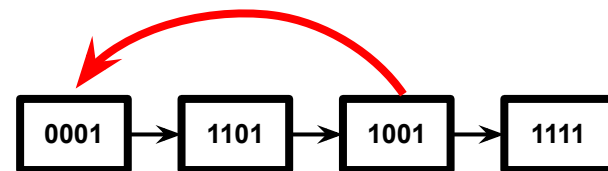
Set Index
1 bit

Block Offset
3 bits

**Use a stack (LIFO) +
hashmap**

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

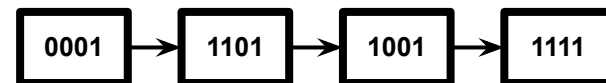
Set Index
1 bit

Block Offset
3 bits

**Use a stack (LIFO) +
hashmap**

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



Tag
4 bits

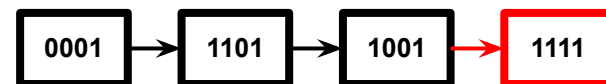
Set Index
1 bit

Block Offset
3 bits

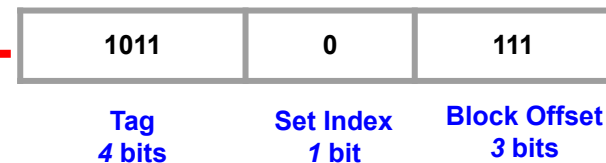
Use a stack (LIFO) +
hashmap

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1111	oooooooo
Set 1	0		
	0		
	0		
	0		



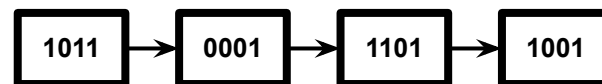
Read Request



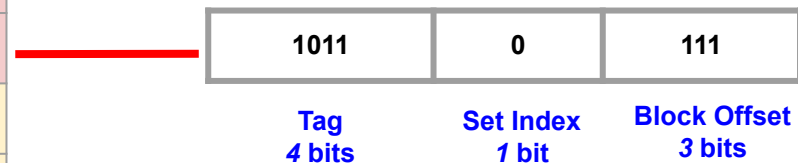
Use a stack (LIFO) +
hashmap

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1011	oooooooo
Set 1	0		
	0		
	0		
	0		



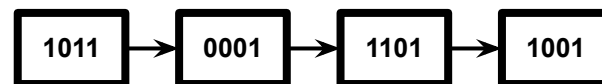
Read Request



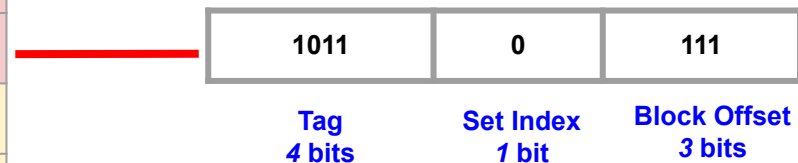
Use a stack (LIFO) +
hashmap

LRU Cache Eviction

	Valid?	Tag	Cache Block (8 bytes)
Set 0	1	1101	oooooooo
	1	0001	oooooooo
	1	1001	oooooooo
	1	1011	oooooooo
Set 1	0		
	0		
	0		
	0		



Read Request



**Use a stack (LIFO) +
hashmap**

Can also use time (global counter)

Performance Metrics

Average memory access time: A rough metric for system memory performance.

$$\text{AMAT} = t_{hit} + \text{rate}_{miss} \times t_{miss}$$

AMAT Example

- 40% of a programs instructions are memory access'
- Cache hit-rate: 90%
- Hit time is 1 cycle, but the miss penalty is 200 cycles.

What is the AMAT on this system?

AMAT Example

- 40% of a programs instructions are memory access'
- Cache hit-rate: 90%
- Hit time is 1 cycle, but the miss penalty is 200 cycles.

What is the AMAT on this system?

$$\begin{aligned} \text{AMAT} &= 1 + (0.10 \times 200) \\ &= 21 \text{ cycles} \end{aligned}$$

PA5 Review

In-class discussion