



CS 1550

Project 3

TA group:

Maher Khan (maherkhan@pitt.edu)	Xiaoyu Liang (xil160@pitt.edu)
Prathamesh Marathe (PSM22@pitt.edu)	Jinpeng Zhou (jiz150@pitt.edu)

Project 3 - Virtual Memory Simulator

- No need to use qemu
- You will write the simulator from scratch with Java, c++,Perl, or Python
- Read from memory traces text files
- Count the number of events (pagefaults, page evictions, hits etc.)
 - Compare eviction algorithms

Project 3 - Virtual Memory Simulator

- Simulate memory page allocation and page eviction algorithm
 - Your program will read from a memory trace
 - You will implement how loaded pages are evicted

Access type:

Load (l) Store (s)

Virtual Address

CPU cycles since
last memory access

New Format:

1 190a7c20 1
s 3856bbe0 1
1 190afc20 1
1 15216f00 1
1 190a7c20 1
1 190a7c28 1
1 190a7c28 1
1 190aff38 1

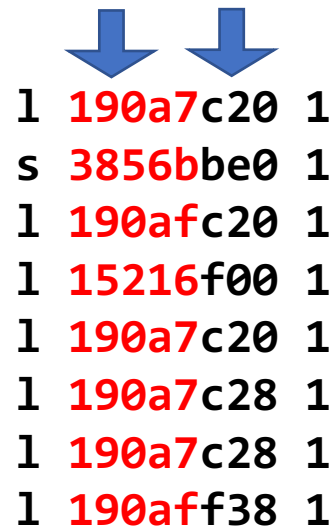
Old Format:

190a7c20 R
3856bbe0 W
190afc20 R
15216f00 R
190a7c20 R
190a7c28 R
190a7c28 R
190aff38 R

Project 3 - Virtual Memory Simulator

- 32-bit address
- 4KB page size (4K: 0x000 ~ 0xFFF)

Page Number Page offset



l	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume FIFO

0	
1	
2	


```
l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1
```

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume FIFO

0	
1	
2	

Pagefault since it is not in the process table




1	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume FIFO

0	190a7
1	
2	

Pagefault since it is not in the process table



1	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume **FIFO**

0	190a7
1	3856b
2	190af

We need to evict
someone!!

Pagefault again

l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume **FIFO**

0	190a7
1	3856b
2	190af

We need to evict
someone!!

Pagefault again



l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 - Virtual Memory Simulator

- Given 3 page frames in 4KB page size
 - Assume **FIFO**

0	3856b
1	190af
2	

We need to evict
someone!!

Pagefault again



l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 - Virtual Memory Simulator

- You need to implement:
 - Opt
 - FIFO
 - Aging

Project 3 – Optimal algorithm

- Evicts the page that will not be used the longest in the future.

Project 3 – Optimal algorithm

- Evicts the page that will not be used the longest in the future.

Pagefault again

0	190a7
1	3856b
2	190af



l **190a7**c20 1
s **3856b**be0 1
l **190af**c20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page that will not be used the longest in the future.

Pagefault again

0	190a7
1	3856b
2	190af

**We need to evict
someone!!**



l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**


Let's analyze who will be needed
furthest away in the trace



0	190a7
1	3856b
2	190af

We need to evict
someone!!

Pagefault again



l	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**

Pagefault again

0	190a7
1	3856b
2	190af

**We need to evict
someone!!**



l **190a7**c20 1
s **3856b**be0 1
l **190af**c20 1
l **15216f00** 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page that will not be used the longest in the future.

Pagefault again

0	190a7
1	3856b
2	190af

We need to evict
someone!!



l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**

Pagefault again

0	190a7
1	3856b
2	190af

**We need to evict
someone!!**

Not used any more



l **190a7**c20 1
s **3856b**be0 1
l **190af**c20 1
l **15216f00** 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**

Pagefault again

0	190a7
1	3856b
2	190af

Evict it.
Since it's a 'Store', write to disk.

l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**

Pagefault again

0	190a7
1	15216
2	190af

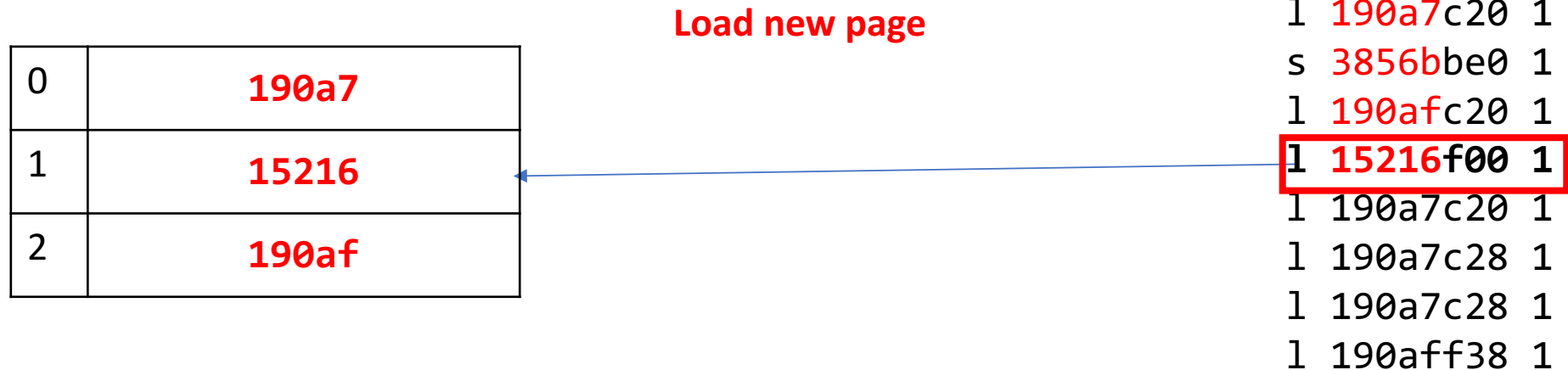
Load new page

l 190a7c20 1
s 385bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – Optimal algorithm

- Evicts the page **that will not be used the longest in the future.**

Pagefault again




Requirement: It should not take more than **5 minutes** to run your program.

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

0	190a7
1	3856b
2	190af

Pagefault again



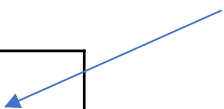
l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1

Project 3 – First In First Out(FIFO)


- Evicts the oldest page in memory.

0	190a7
1	3856b
2	190af

Oldest



Pagefault again



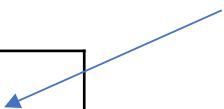
l	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 – First In First Out(FIFO)


- Evicts the oldest page in memory.

0	3856b
1	190af
2	

Oldest



Pagefault again



l	190a7c20	1
s	3856bbe0	1
l	190afc20	1
l	15216f00	1
l	190a7c20	1
l	190a7c28	1
l	190a7c28	1
l	190aff38	1

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	3856b
1	190af
2	15216


l **190a7**c20 1
s **3856b**be0 1
l **190af**c20 1
l **15216f00** 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1



Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	3856b	 Next to be evicted
1	190af	
2	15216	

```
l 190a7c20 1
s 3856bbe0 1
l 190afc20 1
l 15216f00 1
l 190a7c20 1
l 190a7c28 1
l 190a7c28 1
l 190aff38 1
```

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	10000000	190a7
1			
2			

New-loaded page




190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	10000000	190a7
1	0	10000000	3856b
2			



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	1	10000000	190a7
1	0	10000000	3856b
2			



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	1	10000000	190a7
1	0	10000000	3856b
2			

Before this access,
there're already 10 cycles,
refresh first



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	11000000	190a7
1	0	01000000	3856b
2			

Already 10 cycles,
refresh first



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	11000000	190a7
1	0	01000000	3856b
2	0	10000000	190af

New-loaded



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	11000000	190a7
1	0	01000000	3856b
2	0	10000000	190af

Another 10 cycles,
refresh first



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	0 1 100000	190a7
1	0	00 1 00000	3856b
2	0	0 1 000000	190af

Another 10 cycles,
refresh first



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	01100000	190a7
1	0	00100000	3856b
2	0	01000000	190af

Lowest Counter value.
Evict it.



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	0 1 100000	190a7
1	0	00 1 00000	3856b
2	0	0 1 000000	190af



190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11

In this project, evict the **lowest numbered page** (smallest virtual page number) if there're multiple pages with the same lowest counter value. (Real-world: random)

Project 3 – Aging

- Evicts pages that has the lowest counter value.
 - Periodically reduce the counter (right shift by 1 bit)
 - On reference, set leftmost bit of a counter

Assume refresh parameter is 10

	Referenced	Counter (8 bits)	
0	0	01100000	190a7
1	0	10000000	15216
2	0	01000000	190af

New-loaded



```
190a7c20 R 0
3856bbe0 W 3
190a7c20 R 1
190afc20 R 10
15216f00 R 11
```

Project 3 – Program interface

- Program UI

```
./vmsim -n <numframes> -a <opt|aging|fifo> [-r <refresh>] <tracefile>
```

Project 3 – Program interface

- Program UI


```
./vmsim -n <numframes> -a <opt|aging|fifo> [-r <refresh>] <tracefile>
```

Specifies the number of Memory slots.

Project 3 – Program interface

- Program UI

```
./vmsim -n <numframes> -a <opt|aging|fifo> [-r <refresh>] <tracefile>
```




Specifies which algorithm to run

Project 3 – Program interface

- Program UI

```
./vmsim -n <numframes> -a <opt|aging|fifo> [-r <refresh>] <tracefile>
```



Specifies the periodicity of the
refresh rate for the **aging algorithm**

Project 3 – Program interface

- Program UI

```
./vmsim -n <numframes> -a <opt|aging|fifo> [-r <refresh>] <tracefile>
```

Path to memory trace file

CS 1550 – Project 3

- **Due:** Sunday, March 24, 2019 @11:59pm
- **Late:** Tuesday, March 26, 2019 @11:59pm
 - 10% reduction per late day