



CS 1550

Week 9 – Project 3

Teaching Assistant
Xiaoyu(Veronica) Liang

Project 3 - Virtual Memory Simulator

- **NO** need to use QEMU
- You will write the simulator from scratch with **C/C++, Java, 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

Example of a Trace File:

<i>Access type: Load (l) Store (s)</i>	<i>Virtual Address</i>	<i>CPU cycles since last memory access</i>
s	0x1ffffff58	1
l	0x20000060	1
l	0x3004d960	6
l	0x201bd8b0	2
l	0x1ffffff98	2
l	0x1ffffffa0	2
l	0x3004dfa8	2
l	0x20000710	2

Three columns separated by a single space

Project 3 - Virtual Memory Simulator

- 32-bit address
- 4KB page size (4K: 0x000 ~ 0xFFF)
 - First 20 bits is used for the address
 - The rest is used for offset

Page Address Page offset



s	0x1ffffff58	1
l	0x20000060	1
l	0x3004d960	6
l	0x201bd8b0	2
l	0x1ffffff98	2
l	0x1ffffffa0	2
l	0x3004dfa8	2
l	0x20000710	2

Project 3 - Virtual Memory Simulator

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

0	1ffff
1	
2	

Pagefault since it is not
in the process table

➔ s 0x1fffff58 1
l 0x20000060 1
l 0x3004d960 6
l 0x201bd8b0 2
l 0x1ffffff98 2
l 0x1fffffa0 2
l 0x3004dfa8 2
l 0x20000710 2

Project 3 - Virtual Memory Simulator

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

0	1ffff
1	20000
2	

Pagefault



```
s 0x1fffff58 1
l 0x20000060 1
l 0x3004d960 6
l 0x201bd8b0 2
l 0x1ffffff98 2
l 0x1ffffffa0 2
l 0x3004dfa8 2
l 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

0	1ffff
1	20000
2	3004d

Pagefault



```
s 0x1fffff58 1
l 0x20000060 1
l 0x3004d960 6
l 0x201bd8b0 2
l 0x1ffffff98 2
l 0x1ffffffa0 2
l 0x3004dfa8 2
l 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

0	1ffff
1	20000
2	3004d

*We need to evict
someone!!*

Pagefault



```
s 0x1fffff58 1
l 0x20000060 1
l 0x3004d960 6
l 0x201bd8b0 2
l 0x1ffffff98 2
l 0x1ffffffa0 2
l 0x3004dfa8 2
l 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

0	20000
1	3004d
2	

Pagefault



```
s 0x1fffff58 1
l 0x20000060 1
l 0x3004d960 6
l 0x201bd8b0 2
l 0x1fffff98 2
l 0x1fffffa0 2
l 0x3004dfa8 2
l 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

0	20000
1	3004d
2	201bd

Load new page



Pagefault

```
s 0x1fffff58 1
1 0x20000060 1
1 0x3004d960 6
1 0x201bd8b0 2
1 0x1fffff98 2
1 0x1fffffa0 2
1 0x3004dfa8 2
1 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

0	20000
1	3004d
2	201bd

*Next to be evicted,
if need*

Pagefault



```
s 0x1fffff58 1
1 0x20000060 1
1 0x3004d960 6
1 0x201bd8b0 2
1 0x1fffff98 2
1 0x1fffffa0 2
1 0x3004dfa8 2
1 0x20000710 2
```

Project 3 - Virtual Memory Simulator

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

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:** Friday, November 08, 2019 @11:59pm
- **Late:** Sunday, November 10, 2019 @11:59pm
 - 10% reduction per late day