CS 1550

Week 10

–

Project 3

Teaching Assistant

Maher Khan

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

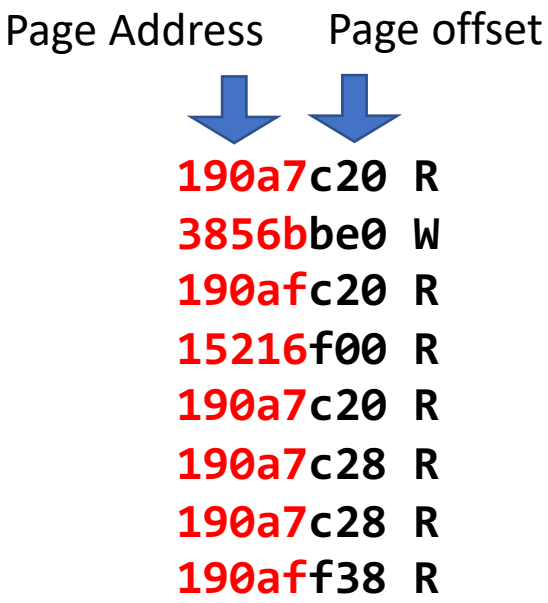
Project 3 - Virtual Memory Simulator

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

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

Project 3 - Virtual Memory Simulator

- Since it is a 32-bit address space.
  - First 20 bits is used for the address
  - The rest is used for offset



Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume FIFO

0	
1	
2	

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

Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume FIFO

0	
1	
2	

Pagefault since it is not in the process table

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

Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume FIFO

0	**190a7**
1	
2	

Pagefault since it is not in the process table

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

Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume **FIFO**

Pagefault again

We need to evict someone!!

0	**190a7**
1	**3856b**
2	**190af**

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

Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume **FIFO**

Pagefault again

0	**190a7**
1	**3856b**
2	**190af**

We need to evict someone!!

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

Project 3 - Virtual Memory Simulator

- Lets suppose you have 12KB of physical memory
  - Page has 4KB
  - Assume **FIFO**

Pagefault again

0	**3856b**
1	**190af**
2	

We need to evict someone!!

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

Project 3 - Virtual Memory Simulator

- You have to implement:
  - Opt (Clairvoyant)
  - FIFO
  - NRU
  - Clock

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**

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

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!!

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

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!!

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

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

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

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!!

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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1	**3856b**	
2	**190af**	

We need to evict someone!!

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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1	**3856b**	4
2	**190af**	

We need to evict someone!!

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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1	**3856b**	4
2	**190af**	3

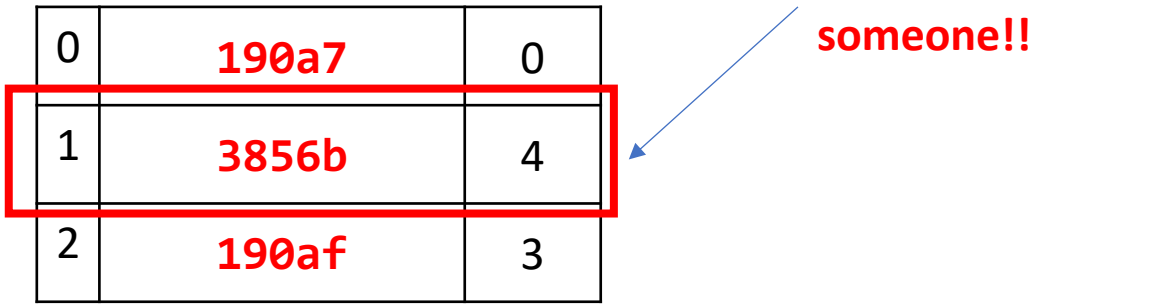
We need to evict someone!!

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

Project 3 – Optimal algorithm

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

Pagefault again

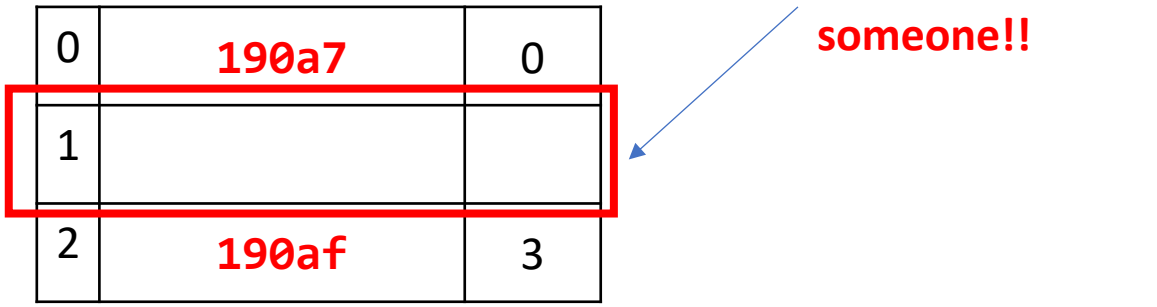



```
190a7c20 R
3856bbe0 W
190afc20 R
15216f00 R
190a7c20 R
190a7c28 R
190a7c28 R
190aff38 R
3856bbe0 R
```

Project 3 – Optimal algorithm

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

Pagefault again




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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1		
2	**190af**	3

We need to evict someone!!

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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1	**15216**	
2	**190af**	3

We need to evict someone!!

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

Project 3 – Optimal algorithm

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

Pagefault again

0	**190a7**	0
1	**15216**	
2	**190af**	3

Remember that this will change as the memory trace progresses

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

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	**190a7**
1	**3856b**
2	**190af**

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

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	**190a7**
1	**3856b**
2	**190af**

We need to evict someone!!

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

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	**3856b**
1	**190af**
2	

We need to evict someone!!

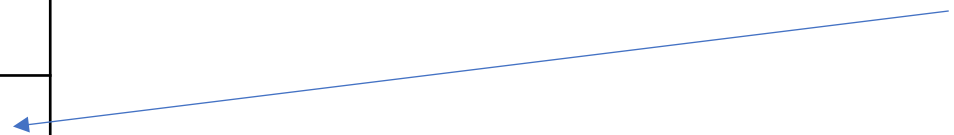
```
190a7c20 R
3856bbe0 W
190afc20 R
15216f00 R
190a7c20 R
190a7c28 R
190a7c28 R
190aff38 R
```

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	**3856b**
1	**190af**
2	**15216**



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

Project 3 – First In First Out(FIFO)

- Evicts the oldest page in memory.

Pagefault again

0	**3856b**
1	**190af**
2	**15216**

Next to be evicted

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to Use and Modification

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

0	**190a7**
1	**3856b**
2	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

	Referenced	**D**irty	
0	1		**190a7**
1	1		**3856b**
2	1		**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**
- Page Rank (Lower Rank -> higher chance of being evicted)

Rank	Page State
3	referenced, modified
2	referenced, **not** modified
1	not referenced, modified
0	not referenced, **not** modified

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

Rank	Page State
3	referenced, modified
2	referenced, **not** modified
1	not referenced, modified
0	not referenced, **not** modified

	Referenced	Dirty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

Rank	Page State
3	referenced, modified
2	referenced, **not** modified
1	not referenced, modified
0	not referenced, **not** modified

	Referenced	Dirty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

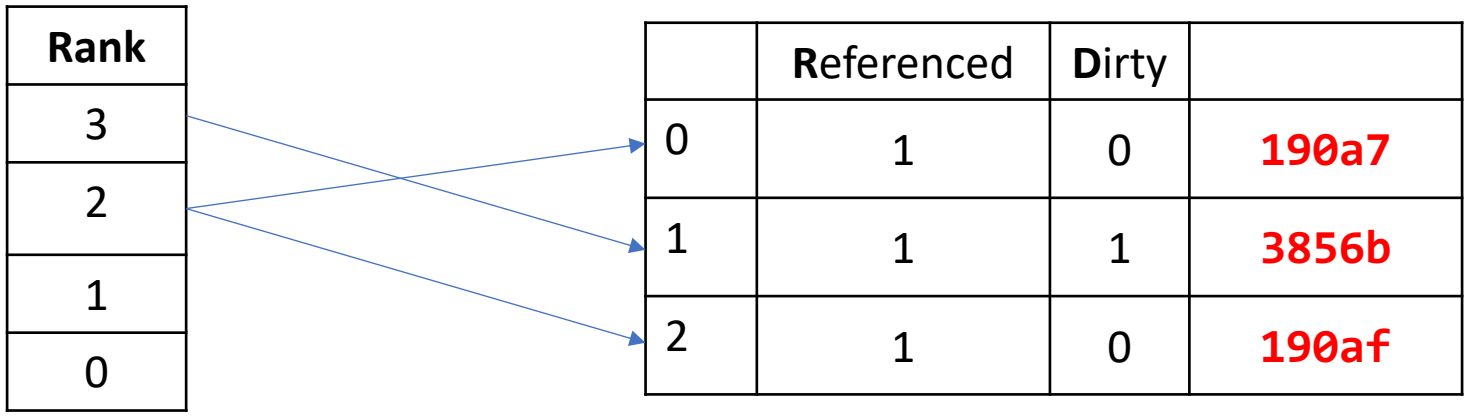
Rank	Page State
3	referenced, modified
2	referenced, **not** modified
1	not referenced, modified
0	not referenced, **not** modified

	Referenced	Dirty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**



Rank
3
2
1
0

	Referenced	Dirty	
0	1	0	190a7
1	1	1	3856b
2	1	0	190af

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

Pagefault

We need to evict someone!! But there are 2 Rank 2 pages.

Rank
3
2
1
0

	Referenced	Dirty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

We need to evict someone!! But there are 2 Rank 2 pages. Chose at random between lowest rank

Rank
3
2
1
0

	Referenced	Dirty	
0	1	0	190a7
1	1	1	3856b
2	1	0	190af

Pagefault

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - Periodically resets referenced flag.
  - Pages are ranked according to **Use** and **Modification**

We need to evict someone!! But there are 2 Rank 2 pages. Chose at random between lowest rank

Rank
3
2
1
0

	Referenced	Dirty	
0	1	0	190a7
1	1	1	3856b
2	1	0	190af

Pagefault

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - **Periodically resets referenced flag**.
  - Pages are ranked according to Use and Modification

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - **Periodically resets referenced flag**.
  - Pages are ranked according to Use and Modification

Has to be set to 0 **periodically** (e.g., every 10 memory accesses)

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Not Recently Used (NRU)

- Evicts pages that are not being used.
  - **Periodically resets referenced flag**.
  - Pages are ranked according to Use and Modification

Has to be set to 0 **periodically** (e.g., every 10 memory accesses)

	Referenced	**D**irty	
0	0	0	**190a7**
1	0	1	**3856b**
2	0	0	**190af**

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	1	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	**190a7**
1	1	1	**3856b**
2	1	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	**190a7**
1	0	1	**3856b**
2	1	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	**190a7**
1	0	1	**3856b**
2	0	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	**190a7**
1	0	1	**3856b**
2	0	0	**190af**

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	**190a7**
1	0	1	**3856b**
2	0	0	**190af**

R is 0 evict!

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	0	0	
1	0	1	**3856b**
2	0	0	**190af**

R is 0 evict!

Pagefault

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

Project 3 – Clock

- Keeps a pointer to the last examined page in a circular list
  - Pointer resets the Referenced bit to 0
    - If R is 0 the page is evicted
  - Pointer tries to evict and moves to the next page when a page fault occur

	Referenced	**D**irty	
0	1	0	**15216**
1	0	1	**3856b**
2	0	0	**190af**

R is 0 evict!

Pagefault

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

Project 3 – Program interface

- Program UI

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

Project 3 – Program interface

- Program UI

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

Specifies the number of Memory slots.

Project 3 – Program interface

- Program UI

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

Specifies which algorithm to run

Project 3 – Program interface

- Program UI

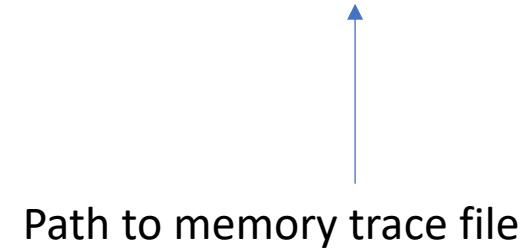
```
./vmsim –n <numframes> -a <opt|clock|fifo|nru> [-r <refresh>] <tracefile>
```

Specifies the periodicity of the refresh rate for the **NRU**

Project 3 – Program interface

- Program UI

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



Project 3 – Program interface

- Program UI

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

```
python vmsim.py    –n 8 -a opt -r ./swim.trace
java   vmsim.class –n 8 -a opt -r ./swim.trace
```

Project 3 – Program interface

- Program UI

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

```
python vmsim.py    –n 8 -a opt -r ./swim.trace
java   vmsim.class –n 8 -a opt -r ./swim.trace
```

Project 3 – Program interface

- As the simulation runs you should print in the following format for **each memory reference**.
  - hit
  - page fault – no eviction
  - page fault – evict clean
  - page fault – evict dirty

Project 3 – Program interface

- As the simulation runs you should print in the following format for **each memory reference**.

```
c:>
```

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

Project 3 – Program interface

- As the simulation runs you should print in the following format for **each memory reference**.

```
c:> python vmsim.py –n 8 -a opt -r ./swim.trace
```

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

Project 3 – Program interface

- As the simulation runs you should print in the following format for **each memory reference**.

```
c:> python vmsim.py –n 8 -a opt -r ./swim.trace

hit
page fault – no eviction
hit
page fault – evict dirty
page fault – evict clean
…
```

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

CS 1550 – Project 3

- **Due**: Friday, March 22, 2019 @11:59pm
- **Late**: Sunday, March 24, 2019 @11:59pm
  - 10% reduction per late day

CS 1550

Week 10

–

Project 3

Teaching Assistant

Maher Khan