

CS 537 Section 6

Michael Swift

3/6/13

© 2004-2007 Ed Lazowska, Hank Levy,
Andrea and Remzi Arpaci-Dusseau, Michael
Swift

1

Questions on Quiz 3

• Question 1

- address space size 512k
- phys mem size 128m
- page size 64k

```
[ 0] 0x80000462
[ 1] 0x00000000
[ 2] 0x80000354
[ 3] 0x80000040
[ 4] 0x00000000
[ 5] 0x80000686
[ 6] 0x00000000
[ 7] 0x00000000
```

- VA 0: 0x00036793
- VA 1: 0x0001d56a
- VA 2: 0x0000cf7d
- VA 3: 0x00028745
- VA 4: 0x00012266

Question 2

- Both Clock and Second-Chance FIFO are approximations of LRU.
 - Pick either the clock algorithm or second-chance FIFO and explain when it makes different decisions than LRU. You can assume that it is run only on-demand (no background paging).
 - How would you change the implementation of clock or second-chance FIFO to make it closer to LRU? How would change then change it to make it **even closer** to LRU?

Question 3

- What is the **copy on write** feature and how is it implemented?
- Under what circumstances is it beneficial to use this feature?
- What hardware support is required to implement this feature?

Questions on Project 3

How does a process get memory?

- Old style:
 - Kernel defines a heap region for a process
 - `brk()` sets the highest address for the heap
 - `sbrk()` adjusts the highest address up or down
 - When heap needs more memory, call `brk()` or `sbrk()` to request more pages
- New style:
 - `mmap(NULL, size, prot, MAP_ANON, -1, 0);`

Real Code

- Linux page frame data base
 - http://lxr.linux.no/linux+v3.8.2/include/linux/mm_types.h
- Flags for each page
 - <http://lxr.linux.no/linux+v3.8.2/include/linux/page-flags.h>
- Linux process control block: `task_struct`
 - <http://lxr.linux.no/linux+v3.8.2/include/linux/sched.h#L1238>

More real code

- Context switch:
 - <http://lxr.linux.no/linux+v3.8.2/kernel/sched/core.c#L1921>
 - http://lxr.linux.no/linux+v3.8.2/arch/x86/include/asm/switch_to.h

System calls

- Table of calls:
 - http://lxr.linux.no/linux+v3.8.2/arch/x86/syscalls/syscall_32.tbl
- Dispatch call:
 - http://lxr.linux.no/linux+v3.8.2/arch/x86/kernel/entry_32.S#L506
- Save registers:
 - http://lxr.linux.no/linux+v3.8.2/arch/x86/kernel/entry_32.S#L186
- Example:
 - <http://lxr.linux.no/linux+v3.8.2/fs/open.c#L970>