CSE3211: Operating System Assignment 0

Al Md. Aladin
student ID: 2015-516-820 & roll number: FH-59

Sharear Bin Amin
student ID: 2015-016-816 & roll number: FH-55

August 13, 2018

1 Questions & Answers

Q1. What is the vm system called that is configured for assignment 0?
Answer: dumbvm from kern/arch/mips/conf/conf.arch

Q2. Which register number is used for the stack pointer (sp) in OS/161?
Answer: #define sp $29 /* stack pointer */ from kern/arch/mips/include/kern/regdefs.

Q3. What bus/busses does OS/161 support?
Answer: LAMEbus from kern/arch/sys161/include/bus.h

Q4. What is the difference between splhigh and spl0?
Answer: splhigh() sets IPL to the highest value, disabling all interrupts. spl0() sets IPL to 0, enabling all interrupts. from os161-ASST0/kern/include/spl.h

Q5. Why do we use typedefs like u_int32_t instead of simply saying "int"?
Answer: To make sure that we really get a 32-bit unsigned integer (unsigned int depends on the platform) from kern/arch/mips/include/types.h

Q6. What must be the first thing in the process control block?
Answer: pcb_switchstack must be the first thing in the process control block.

Q7. What does splx return?
Answer: The old interrupt state (an integer) from os161-ASST0/kern/include/spl.h

Q8. What is the highest interrupt level?
Answer: #define IPL_HIGH 1 from os161-ASST0/kern/include/spl.h

Q9. What function is called when user-level code generates a fatal fault?
Answer: void kill_curthread(vaddr_t epc, unsigned code, vaddr_t vaddr) from os161-ASST0/kern/arch/mips/mips/trap.c

Q10. How frequently are hardclock interrupts generated?
Answer: 100 times a second #define HZ 100 from kern/include/clock.h

Q11. What functions comprise the standard interface to a VFS device?
Answer: vfs_setcurdir, vfs_clearcurdir, vfs_getcurdir, vfs_sync, vfs_getroot, vfs_getdevname, vfs_lookup, vfs_lookparent, vfs_open, vfs_close, vfs_readlink, vfs_symlink, vfs_mkdir, vfs_link, vfs_remove, vfs_rmdir, vfs_rename, vfs_chdir, vfs_getcwd, vfs_bootstrap, vfs_initbootfs, vfs_setbootfs, vfs_clearbootfs, vfs_adddev, vfs_addfs, vfs_mount, vfs_unmount, and vfs_unmountall from os161-ASST0/kern/include/vfs.h

Q12. How many characters are allowed in a volume name?
Answer: #define SFS_VOLNAME_SIZE 32 /* max length of volume name */ from kern/include/kern/sfs.h

Q13. How many direct blocks does an SFS file have?
Answer: #define SFS_NDIRECT 15 /* # of direct blocks in inode */ from kern/include/kern/sfs.h

Q14. What is the standard interface to a file system (i.e., what functions must you implement to implement a new file system)?
Answer: fsop_sync, fsop_getvolname, fsop_getroot, fsop_umount from kern/include/fs.h

Q15. What function puts a thread to sleep?
Answer: Void wchan_sleep(struct wchan *wc, struct spinlock *lk) from kern/thread/thread.c

Q16. How large are OS/161 pids?
Answer: typedef int32_t pid_t; /* Process ID */ 32 bits / 4 bytes from kern/include/kern/types.h

Q17. What operations can you do on a vnode?
Answer: vop_eachopen,vop_ reclaim,vop_ read, vop_readlink, vop_getdirentry, vop_write, vop_ioctl, vop_stat, vop_gettype, vop_tryseek, vop_fsync, vop_mmap, vop_truncate, vop_namefile, vop_creat, vop_symlink, vop_mkdir, vop_link, vop_remove, vop_rmdir, vop_rename, vop_lookup, vop_lookparent from kern/include/vnode.h

Q18. What is the maximum path length in OS/161?
Answer: * Longest full path name */ #define PATH_MAX 1024 from kern/include/kern/limits.h

Q19. What is the system call number for a reboot?
Answer: The system call number for a reboot is 119.

Q20. Where is STDIN_FILENO defined?
Answer: #define STDIN_FILENO 0 /* Standard input */ from kern/include/kern/unistd.h

Q21. What does kmain() do?
Answer: Kernel main. (Boot up, then fork the menu thread, wait for a reboot request, and then shut down.) from kern/main/main.c

Q22. Is it OK to initialise the thread system before the scheduler? Why (not)?
Answer: Yes. The scheduler bootstrap just creates the run queue, and the thread bootstrap just initializes the first thread.

Q23. What is a zombie?
Answer: "Zombies are threads/processes that have exited but not been fully deleted yet." from kern/thread/thread.c

Q24. How large is the initial run queue?
Answer: The initial run queue is 32.

Q25. What does a device name in OS/161 look like?
Answer: The name of a device is always device:, such as lhd0: from kern/vfs/device.c

Q26. What does a raw device name in OS/161 look like?
Answer: The name with raw appended, such as lhd0raw, from

kern/vfs/vfslist.c

Q27. What lock protects the vnode reference count?
Answer: vn_countlock protects the vnode reference count from kern/vfs/vnode.c

Q28. What device types are currently supported?
Answer: The device types currently supported are block and character devices. from kern/vfs/device.c

.....