

EXERCÍCIOS DE INTRODUÇÃO À ARQUITETURA DE COMPUTADORES

Prof. Jorge Habib Hanna El Khouri

Aluno: _____

Turma: 2º A

UNIOESTE

PARTE I: Exercícios sobre Interrupções e I/O:

1. Qual a diferença entre *polling* e *interrupção*?
2. O que vem a ser endereço de *I/O*? Quais os endereços de *I/O* do controlador do teclado, da *PIC*, do *RTC*, das portas seriais *COM1* e *COM2* e da porta paralela *LPT1*?
3. O que é *IRQ*?
4. Quais são as etapas realizadas para tratar uma requisição de um dispositivo de *I/O*?
5. Quais as diferenças nas semânticas de um *jump*, de uma chamada de função, da execução de um processo e de uma interrupção?
6. Como estão classificadas as interrupções no *Pentium*?
7. Quais são as etapas realizadas para tratar uma requisição de um dispositivo de *I/O*?
8. Qual o funcionamento básico do 8259A (*PIC* – Programmable Interrupt Controller)?
9. Explique o que é Interrupção Mascarável e Não-Mascarável, e para que servem as instruções *cli* e *sti*.
10. Explique para que serve o *Vetor de Interrupções (IDT)*:
 - a. Qual a utilização de cada uma das interrupções de 0 a 19?
 - b. Qual a utilização de cada uma das interrupções de 32 a 47?
 - c. Escolha uma do item *a* e uma do item *b* e pesquise detalhes sobre elas no código do Linux.
11. Como você classifica a *Interrupção 0x80* do *Linux*? Que funcionalidades fazem parte desta interrupção?
12. Esboce um diagrama de blocos mostrando a relação entre dispositivos de *I/O*, *PIC*, Processador, *IDT*, *Interrupt Service Routine (ISR)*, ...

PARTE II: Escrever functions em assembly que implementem, através de chamadas aos recursos da *INT 0x80*, as seguintes funcionalidades.

1. O que é um *File Descriptor*?
2. Criar um arquivo;
3. Abrir um arquivo;
4. Fechar um arquivo;
5. Escrever dados da memória para um arquivo;
6. Ler dados de um arquivo para a memória;
7. Remover um arquivo;
8. Escrever uma string na tela;
9. Ler uma string do teclado;
10. Criar um sub-diretório;
11. Renomear um arquivo;
12. Remover um sub-diretório;
13. Obter a data do S.O.;
14. Obter a hora do S.O.;
15. Obter o diretório atual;
16. Alterar o diretório atual;

File Permission								
0	0	0	0	0	0	0	0	0
8	7	6	5	4	3	2	0	0
R	W	X	R	W	X	R	W	X
User			Group			Other		

File Access Mode								
#define O_ACCMODE 00000003								
#define O_RDONLY 00000000								
#define O_WRONLY 00000001								
#define O_RDWR 00000002								
#define O_CREAT 00000100 /* not fcntl */								

17. Implemente em assembly (usando as funções de 1 a 5) uma função para copiar um arquivo:
int copyfile (char * src, char * dst);
18. Pesquise e utilize o comando *strace* do *Linux* para observar o uso das “*system calls*”. Consulte os serviços utilizados pelos comandos *cp*, *cd*, *ls*, *mkdir*, *date* e outros.

INT 0x80		Create and open a file	Open a file	Read from a file	Write to a file	Close a file
INPUT	EAX	8	5	3	4	6
	EBX	filename	filename	file descriptor	file descriptor	file descriptor
	ECX	file permissions	file access mode	pointer to input buffer	pointer to output buffer	
	EDX		file permissions	buffer size (maximum number of bytes to read)	buffer size (number of bytes to write)	
RETURNS	EAX	file descriptor	file descriptor	number of bytes read	number of bytes written	
ERROR	EAX	error code	error code	error code	error code	error code

#	Name	Registers			
		eax	ebx	ecx	edx
1	sys_exit	0x01	int error_code	-	-
2	sys_fork	0x02	struct pt_regs *	-	-
3	sys_read	0x03	unsigned int fd	char __user *buf	size_t count
4	sys_write	0x04	unsigned int fd	const char __user *buf	size_t count
5	sys_open	0x05	const char __user *filename	int flags	int mode
6	sys_close	0x06	unsigned int fd	-	-
8	sys_creat	0x08	const char __user *pathname	int mode	-
9	sys_link	0x09	const char __user *oldname	const char __user *newname	-
10	sys_unlink	0x0a	const char __user *pathname	-	-
12	sys_chdir	0x0c	const char __user *filename	-	-
13	sys_time	0x0d	time_t __user *tloc	-	-
15	sys_chmod	0x0f	const char __user *filename	mode_t mode	-
19	sys_lseek	0x13	unsigned int fd	off_t offset	unsigned int origin
20	sys_getpid	0x14	-	-	-
33	sys_access	0x21	const char __user *filename	int mode	-
37	sys_kill	0x25	int pid	int sig	-
38	sys_rename	0x26	const char __user *oldname	const char __user *newname	-
39	sys_mkdir	0x27	const char __user *pathname	int mode	-
40	sys_rmdir	0x28	const char __user *pathname	-	-
45	sys_brk	0x2d	unsigned long brk	-	-
78	sys_gettimeofday	0x4e	struct timeval __user *tv	struct timezone __user *tz	-
79	sys_settimeofday	0x4f	struct timeval __user *tv	struct timezone __user *tz	-
92	sys_truncate	0x5c	const char __user *path	long length	-
94	sys_fchmod	0x5e	unsigned int fd	mode_t mode	-
162	sys_nanosleep	0xa2	struct timespec __user *rqtp	struct timespec __user *rmtp	-