

ICS22期末答案

By罗兆丰 曹思诺

第一题 填空题

- 1、RISC CISC
- 2、10
- 3、rdi rsi
- 4、符号解析 重定位
- 5、GOT PLT
- 6、① gcc p.o libx.a liby.a ②gcc p.o libx.a liby.a libx.a
- 7、①发生异常，从用户态切换到内核态 ②进程A的信号处理程序
- 8、fork execve
- 9、MMU TLB
- 10、0x5d5
- 11、内存映射
- 12、36.152.44.95
- 13、bind
- 14、对等线程
- 15、互斥锁

第二题 处理器

- (1) E_icode d_srcA d_srcB
- (2) stall bubble normal
- (3) D_icode E_icode M_icode
- (4) bubble normal normal
- (5) 3
- (6) 前一条指令从内存中加载数据到寄存器 `rsp` 中，后面紧跟了一条 `ret` 指令

第三题 链接

(1) 注意，最后一列填的是“此符号在定义该符号的模块中的节”。`addvec`在`addvec.o`中是在`.text`，在`main.o`中是在`UND`中

符号	.symtab条目?	符号类型	定义符号的模块	节或伪节名
x	是	全局	main.o	.data
y	否	/	/	/
z	是	全局	main.o	COMMON
n	否	/	/	/
addvec	是	外部	addvec.o	.text

(2) 20 10 60 00
25 00 00 00

(3) A

`.interp`节的内容其实就是一个字符串，为动态链接器的路径。把`.interp`节的所有字节一个地翻译成字符，就得到`/lib64/ld-linux-x86-64.so.2`。（大家可以特别记一下，'0'的ascii的16进制是0x30）

(4) 1 3

第四题

1. (1) A: 0

B: 1

C: `Sigprocmask`

D: `&mask_all`

E: `Sigsuspend`

F: 1

(2) 是

否则可能会 1) 父进程向子进程发送多个同类信号，而子进程只接受到1个，子进程死循环。2)

子进程处理`SIGUSR1`和`SIGUSR2`的顺序与父进程发信号的顺序不一致。两个情况都会导致出错

2. （这个小题应该是出错了，如果按照原题目的话：）不会输出。在`sigusr_handler`中，如果G为真，执行`return`，那么就不会发送 `SIGCONT`给父进程，父进程阻塞在`Sigsuspend`，所以子进程也会阻塞，导致没有输出。

第五题 虚拟内存管理

1、①代码 ②运行时堆 ③共享库的内存映射区域 ④用户栈

2、触发保护异常，终止进程 触发段错误，终止进程

3、

① 21行 所申请的内存大小不足，应该为`malloc(size*sizeof(int));`

② 27与31行 不能返回局部变量的地址

③ 59行 n所指向的地址在`push`后可能已经被释放

④ 61行 `stack.data`没有被释放，造成内存泄露

第六题 网络编程

(1) SOCK_STREAM

(2) IP地址对应的主机名

(3) localhost 50000

(4) 每次循环都打开了一个新的clientfd而没有关闭，导致文件数量过多，耗尽系统资源

(5) 22 Close(clientfd);

(6) A 小明的服务器在迭代处理两个客户端时，bal_cnt只会在某一次与其中一个客户端连接时等于goal，之后便会设为0，而另一个客户端已经超过0，因此只会发送一次“You win!”

第七题 并发

第一类:

- | | | | |
|---------------|---------------|---------------|---------------|
| 1、 P(&mutex); | 2、 P(&w); | 3、 V(&mutex); | 4、 P(&mutex); |
| 5、 V(&w); | 6、 V(&mutex); | 7、 P(&w); | 8、 V(&w); |

第二类:

- | | | | |
|--------------------------|---------------------------|---------------------------|-----------|
| 1、 1 | 2、 P(&r); | 3、 if(readcnt==1) P(&w); | 4、 V(&r); |
| 5、 if(readcnt==0) V(&w); | 6、 if(writecnt==1) P(&r); | 7、 if(writecnt==0) V(&r); | |