

# 2020 春 第 8~9 章 家庭作业

最后一次作业啦，一定要按时提交哦~

第 8 章作业：第 2 版教材 P528~529 8.18 & 8.23

第 9 章作业：第 2 版教材 P586~587 9.11 & 9.12

要求：一定要有写分析过程，不要光填表啊！

8.18 考虑下面的程序：

```
code/ecf/forkprob2.c
1  #include "csapp.h"
2
3  void end(void)
4  {
5      printf("2");
6  }
7
8  int main()
9  {
10     if (Fork() == 0)
11         atexit(end);
12     if (Fork() == 0)
13         printf("0");
14     else
15         printf("1");
16     exit(0);
17 }
```

code/ecf/forkprob2.c

判断下面哪个输出是可能的。注意：atexit 函数以一个指向函数的指针为输入，并将它添加到函数列表中（初始为空），当 exit 函数被调用时，会调用该列表中的函数。

- A. 112002
- B. 211020
- C. 102120
- D. 122001
- E. 100212

8.23 你的一个同事想要使用信号来让一个父进程对发生在一个子进程中的事件计数。其思想是每次发生一个事件时，通过向父进程发送一个信号来通知它，并且让父进程的信号处理程序对一个全局变量 counter 加一，在子进程终止之后，父进程就可以检查这个变量。然而，当他在系统上运行图 8-41 中的测试程序时，发现当父进程调用 printf 时，counter 的值总是 2，即使子进程向父进程发送了 5 个信号。他很困惑，向你寻求帮助。你能解释这个程序有什么错误吗？

```

1  #include "csapp.h"
2
3  int counter = 0;
4
5  void handler(int sig)
6  {
7      counter++;
8      sleep(1); /* Do some work in the handler */
9      return;
10 }
11
12 int main()
13 {
14     int i;
15
16     Signal(SIGUSR2, handler);
17
18     if (Fork() == 0) { /* Child */
19         for (i = 0; i < 5; i++) {
20             Kill(getppid(), SIGUSR2);
21             printf("sent SIGUSR2 to parent\n");
22         }
23         exit(0);
24     }
25
26     Wait(NULL);
27     printf("counter=%d\n", counter);
28     exit(0);
29 }

```

---

co

图 8-41 家庭作业 8.23 中引用的计数器程序

9.11 在下面的一系列问题中，你要展示 9.6.4 节中的示例存储器系统如何将虚拟地址翻译成物理地址，以及如何访问缓存。对于给定的虚拟地址，请指出访问的 TLB 条目、物理地址，以及返回的缓存字节值。请指明是否 TLB 不命中，是否发生了缺页，是否发生了缓存不命中。如果有缓存不命中，对于“返回的缓存字节”用“-”来表示。如果有缺页，对于“PPN”用“-”来表示，而 C 部分和 D 部分就空着。

虚拟地址：0x027c

A. 虚拟地址格式

13	12	11	10	9	8	7	6	5	4	3	2	1	0

B. 地址翻译

参数	值
VPN	
TLB 索引	
TLB 标记	
TLB 命中？（是 / 否）	
缺页？（是 / 否）	
PPN	

C. 物理地址格式

11	10	9	8	7	6	5	4	3	2	1	0

D. 物理地址引用

参数	值
字节偏移	
缓存索引	
缓存标记	
缓存命中？（是 / 否）	
返回的缓存字节	

9.12 对于下面的地址，重复习题 9.11：

虚拟地址：0x03a9

A. 虚拟地址格式

13	12	11	10	9	8	7	6	5	4	3	2	1	0

B. 地址翻译

参数	值
VPN	
TLB 索引	
TLB 标记	
TLB 命中？（是 / 否）	
缺页？（是 / 否）	
PPN	

C. 物理地址格式

11	10	9	8	7	6	5	4	3	2	1	0

D. 物理地址引用

参数	值
字节偏移	
缓存索引	
缓存标记	
缓存命中？（是 / 否）	
返回的缓存字节	

PS.

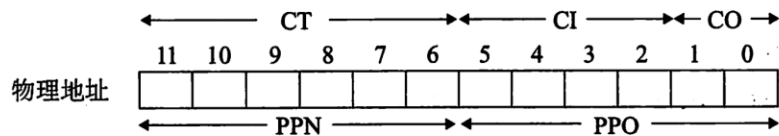
以下为第 2 版教材 9.6.4 节中示例的相关存储系统内容，作业可能会用到。

位	标记位	PPN	有效位	标记位	PPN	有效位	标记位	PPN	有效位	标记位	PPN	有效位
0	03	-	0	09	0D	1	00	-	0	07	02	1
1	03	2D	1	02	-	0	04	-	0	0A	-	0
2	02	-	0	08	-	0	06	-	0	03	-	0
3	07	-	0	03	0D	1	0A	34	1	02	-	0

a) TLB: 四组, 16 个条目, 四路组相联

VPN	PPN	有效位	VPN	PPN	有效位
00	28	1	08	13	1
01	—	0	09	17	1
02	33	1	0A	09	1
03	02	1	0B	-	0
04	—	0	0C	-	0
05	16	1	0D	2D	1
06	—	0	0E	11	1
07	—	0	0F	0D	1

b) 页表: 只展示了前 16 个 PTE



索引 标记位 有效位 块 0 块 1 块 2 块 3

0	19	1	99	11	23	11
1	15	0	—	—	—	—
2	1B	1	00	02	04	08
3	36	0	—	—	—	—
4	32	1	43	6D	8F	09
5	0D	1	36	72	F0	1D
6	31	0	—	—	—	—
7	16	1	11	C2	DF	03
8	24	1	3A	00	51	89
9	2D	0	—	—	—	—
A	2D	1	93	15	DA	3B
B	0B	0	—	—	—	—
C	12	0	—	—	—	—
D	16	1	04	96	34	15
E	13	1	83	77	1B	D3
F	14	0	—	—	—	—

c) 高速缓存: 16 个组, 4 字节的块, 直接映射