

# 操作系统 2022 答案

Deschain

2022 年 7 月 11 日

一、

1.C 2.B 3.B 4.D 5.A 6.B 7.D 8.D 9.C 10.A

二、

1.

运行态 0 4 个，就绪态 0 6 个，阻塞态 0 10 个。

2.

$$10ms + 60s \div 5400 \div 2 = 15.6ms$$

3.

共有  $20GB \div 4KB = 5M$  个簇，因此位图大小为 5M，需要  $5M \div 4KB = 5M \div 32K = 160$  个簇来存放。

4.

$$\text{单缓冲区: } (100 + 10) \times 100 = 11000\mu s = 11ms$$

$$\text{双缓冲区: } (100 + 10) \times 100 = 11000\mu s = 11ms$$

5.

$$\frac{20}{100} + \frac{40}{150} + \frac{100}{300} = 0.8 < 1, \text{ 可调度}$$

6.

$16 \times 0.5ms = 8ms$ ，中断处理方式是合理的，因为鼠标需要的时间占比很小。

7.

$$\lfloor \frac{12345}{4 \times 1024} \rfloor = 3, \text{ 页内偏移量 } 57, \text{ 物理地址 } 6 \times 4 \times 1024 + 57 = 24633$$

8.

$$\lfloor \frac{46000}{2 \times 1024} \rfloor = 22, \text{ 块内偏移量 } 944, \text{ 位于第 } 92 + 22 = 114 \text{ 块。}$$

三、

1.

FCFS	A	A	A	A	A	A	B	B	B	C	C	C	C	D	D	D	D	D	E	E
SJF	A	B	B	B	C	E	E	C	C	C	D	D	D	D	D	A	A	A	A	A
HRRN	A	A	A	A	A	A	B	B	B	E	E	C	C	C	C	D	D	D	D	D
SRT	A	B	B	B	C	E	E	C	C	C	A	A	A	A	A	D	D	D	D	D

周转时间：

	A	B	C	D	E	系统
FCFS	6	8	11	14	15	10.8
SJF	20	3	8	11	2	8.8
HRRN	6	8	13	16	6	9.8
SRT	15	3	8	16	2	8.8

2.

OPT	M	M	M	H	M	H	M	H	H	M	H	H	6 次
FIFO	M	M	M	H	M	H	M	M	M	M	H	M	9 次
LRU	M	M	M	H	M	M	M	M	H	M	H	M	9 次
CLOCK	M	M	M	H	M	M	M	M	H	M	H	M	9 次

3.

(a)  $256^2 \times 2 \times 100ns = 13107200ns = 13.1072ms$

(b) 共用到  $256^2 \times 4 / (4 \times 1024) = 64$  页，初次访问缺页，因此 TLB 的命中率为  $1 - 64 / (256^2) = 99.902\%$

(c)  $64 - 16 = 48$  次

(d)  $256^2 \times (10ns + 100ns) + 64 \times 100ns = 7215360ns = 7.21536ms$

4.

```
typedef Semaphore int;
```

```
Semaphore Guard = 1; //保安空闲
```

```
Semaphore GuardQueue = 0; //等待保安扫码的人数
```

```
Semaphore Inside = 5; //储蓄所可进人数
```

```
Semaphore Counter = 1; //柜员空闲
```

```
Semaphore CounterQueue = 0; //等待柜员服务的人数
```

```
Semaphore door = 1; //闸机空闲
```

```
void CustomerFunc(){
```

```
V(GuardQueue); //通知保安，有人等待扫码
```

```
P(Guard); //申请保安服务
```

```
扫码;
```

```
P(Inside); //在入口处申请进门
```

```
进入储蓄所;
```

```
V(CounterQueue); //通知柜员，有人等待服务
```

```
P(Counter); //申请柜员服务
```

```
接受柜员服务;
```

```
P(Door); // 申请使用闸机
出门;
V(Door);
V(Inside);
}
void GuardFunc(){
P(GuardQueue);
扫码;
V(Guard);
}
void CounterFunc(){
P(CounterQueue);
服务;
V(Counter);
}
```