

《操作系统大题》

1、在一个批处理单道系统中，采用响应比高者优先的作业调度算法。当一个作业进入系统后就可以开始调度，假定作业都仅是计算，忽略调度花费的时间。现有 3 个作业，进入系统的时间和需要计算的时间如表 2-2 所示。

表 2-2 进入系统的时间和需要计算的时间表

作业	进入系统时间	需要计算时间	开始时间	完成时间	周转时间
1	9:00	60 分钟	—	—	—
2	9:10	45 分钟	—	—	—
3	9:15	25 分钟	—	—	—

(1) 求出每个作业的开始时间、完成时间及周转时间。

解答：

先来先服务：

作业	进入系统时间	需要计算时间	开始时间	完成时间	周转时间
1	9:00	60 分钟	9:00	10:00	60 分钟
2	9:10	45 分钟	10:00	10:45	95 分钟
3	9:15	25 分钟	10:45	11:10	115 分钟

响应比高者优先：

作业	进入系统时间	需要计算时间	开始时间	完成时间	周转时间
1	9:00	60 分钟	9:00	10:00	60 分钟
2	9:10	45 分钟	10:25	11:10	120 分钟
3	9:15	25 分钟	10:00	10:25	70 分钟

短作业优先：

作业	进入系统时间	需要计算时间	开始时间	完成时间	周转时间
1	9:00	60 分钟	9:00	10:00	60 分钟
2	9:10	45 分钟	10:25	11:10	120 分钟
3	9:15	25 分钟	10:00	10:25	70 分钟

(2) 计算三个作业的平均周转时间应为多少？

解答：

先来先服务：

$$(60+95+115) / 3 = 90(\text{分钟})$$

响应比高者优先：

$$(60+120+70) / 3 = 83.33(\text{分钟})$$

短作业优先：

$$(60+120+70) / 3 = 83.33(\text{分钟})$$

2、一个计算机系统有 6 个磁带驱动器和 n 个进程。每个进程最多需要两个磁带驱动器。问当 n 为什么值时，系统不会发生死锁？

解答：

方法一：对于 3 个进程，每个进程能够有两个驱动器。对于 4 个进程，驱动器可以按照 (2,2,1,1) 的方法进行分配，使前面两个进程先结束。对于 5 个进程，可以按照 (2,1,1,1,1) 的方式进行分发，使一个进程先结束。对于 6 个进程，每个进程都拥有一个磁带驱动器同时需要另外一个驱动器，产生了死锁。因此，

对于 $n < 6$ 的系统来说是无锁的。

方法二：已知系统中的每个进程需要 2 个驱动器。那么在最坏的情况下，各进程都占用了其中的一个，而且都在请求自己所需要的另一个。如果此时系统尚有多于一个，那么就可以满足其中一个进程运行完毕。当该进程运行完毕释放出它所有占有的驱动器后，又可进一步满足其他进程。系统不会出现死锁。因此，如果将 $(6-1)$ 个驱动器机分配给 n 个进程，满足每个进程一个的话，进程数量 n 必然小于等于 5，此时系统中不会发生死锁。

3、假定某磁盘共有 200 个柱面，编号为 0~199，如果在为访问 143 号柱面的请求者服务后，当前正在为访问 125 号柱面的请求者服务，同时有若干请求者在等待服务，它们依次要访问的柱面号为：86, 147, 91, 177, 94, 150, 102, 175, 130，请回答下列问题：

(1) 分别用先来先服务算法、最短寻找时间优先算法、电梯调度算法和单向扫描算法来确定实际的服务次序。

解答：143→125

先来先服务算法：86→147→91→177→94→150→102→175→130

最短寻找时间优先算法（离 125 最近）：130→147→150→175→177→102→94→91→86

电梯调度算法：102→94→91→86→130→147→150→175→177

单向扫描算法：130→147→150→175→177→86→91→94→102

(3) 按实际服务次序计算上述算法下移动臂需移动的距离。

解答：

先来先服务：

$(125-86) + (147-86) + (147-91) + (177-91) + (177-94) + (150-94) + (150-102) + (175-102) + (175-130) = 547$

最短寻找时间优先：

$(130-125) + (147-130) + (150-147) + (175-150) + (177-175) + (177-102) + (102-94) + (94-91) + (91-86) = 143$

电梯调度算法： $(125-102) + (102-94) + (94-91) + (91-86) + (130-86) + (147-130) + (150-147) + (175-150) + (177-175) = 130$

单向扫描算法： $(130-125) + (147-130) + (150-147) + (175-150) + (177-175) + (177-86) + (91-86) + (94-91) + (102-94) = 154$ （除移动臂返回的时间外）