

3rd CPU 调度

1. Importance of distinguishing I/O-bound programs from CPU-bound programs?

A: I/O密集型程序通常具有大量短CPU执行，CPU密集型程序可能只有少量长CPU执行。区分两种类型的程序，对于选择合适的CPU调度算法从而提高运行效率有重要作用。

2. a. ① FCFS:

	P ₁	P ₂	P ₃	P ₄	P ₅	
0		10	11	13	14	19

② SJF:

P ₂	P ₄	P ₃	P ₅	P ₁	
0	1	2	4	9	19

③ nonpreemptive priority (smaller \rightarrow higher)

P ₂	P ₅	P ₁	P ₃	P ₄
0	1	6	16	18

④ RR (quantum = 1)

P ₁	P ₂	P ₃	P ₄	P ₅	P ₁	P ₃	P ₁	P ₅	P ₁	P ₂	P ₁	P ₄	
0													19

b. 周转时间 (等待 + 执行)

process	FCFS	SJF	nonpreemptive priority	RR
P ₁	10	19	16	19
P ₂	11	1	1	2
P ₃	13	4	18	7

(接上页表)

process	FCFS	SJF	nonpreemptive priority	RR
P ₄	14	2	19	4
P ₅	19	9	6	14
Avg	13.4	7	12	9.2

c. 等待时间

process	FCFS	SJF	nonpreemptive priority	RR
P ₁	0	9	6	9
P ₂	10	0	0	1
P ₃	11	2	16	5
P ₄	13	1	18	3
P ₅	14	4	1	9
Avg	9.6	3.2	8.2	5.4

d. 短作业优先(SJF) 算法平均等待时间为 3.2 (ms)

3. Which of the following scheduling algorithms could result in starvation?

b. short job first (SJF) and d. Priority

4. A: (1) new priorities:

$$P_1 = 40/2 + 60 = 80, P_2 = 18/2 + 60 = 69, P_3 = 10/2 + 60 = 65$$

优先级从高到低为 P_3, P_2, P_1

(2) 传统的 unix 调度方法降低了 CPU 密集型进程的相对优先级。

因为 CPU 密集型进程比 I/O 密集型进程占用更多 CPU 时间，该计算方法以 Recent CPU usage / 2 为增量，CPU 密集型进程 Priority 值增长更快，故优先级也降低得更快，所以其相对优先级是降低的。