

操作系统(1)

第三章 处理及调度与死锁 课程作业

严昕宇 20121802

上海大学 计算机工程与科学学院

1 假设一个系统中有5个进程,它们的到达时间和服务时间如表3-1所示,忽略I/O以及其他开销时间,若分别按先来先服务(FCFS)、非抢占及抢占的短作业优先(SJF)、高响应比优先(HRRN)、时间片轮转(RR,时间片=1)、多级反馈队列调度算法(FB,第*i*级队列的时间片= 2^{i-1})进行CPU调度,请给出各进程的完成时间、周转时间、带权周转时间、平均周转时间和平均带权周转时间。

表 3-1 进程到达和需服务时间

进程	到达时间	服务时间
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

1.1 先来先服务(FCFS, First Come First Service)

按照到达顺序,依次运行计算即可,结果如下:

进程名	A	B	C	D	E	平均
完成时间	3	9	13	18	20	
周转时间	3	7	9	12	12	8.6
带权周转时间	1	1.17	2.25	2.4	6	2.56

1.2 非抢占式短作业优先(SJF, Shortest Job First)

在0时刻时,只有进程A,因此首先运行进程A。进程A在3时刻执行完毕,此时只有进程B到达,因此运行进程B。9时刻时,进程B执行完毕,此时进程C、D、E皆到达,由于使用SJF算法,其中进程E的服务时间最短,因此优先运行进程E。进程E在11时刻执行完毕,进程C、D中C的服务时间最短,因此优先运行进程C。进程C在14时刻执行完毕,运行进程D,并在20时刻最终执行完所有作业。具体数据如下:

进程名	A	B	C	D	E	平均
完成时间	3	9	15	20	11	
周转时间	3	7	11	14	3	7.6
带权周转时间	1	1.17	2.75	2.8	1.5	1.84

1.3 抢占式短作业优先(SRTN, Shortest Remaining Time Next)

在0时刻时，只有进程A，因此首先运行进程A。2时刻进程B已经到达，且服务时间长于A(1)，因此继续运行进程A。3时刻时，进程A执行完毕，运行进程B。4时刻时，进程C(4)达到，服务时间小于目前进程B(5)，因此先运行进程C。6时刻时，进程D(5)到达，但其服务时间大于进程C(2)，因此继续执行进程C。8时刻时，进程C已经执行完毕，进程E到达，服务B、D、E中进程E(2)的服务时间最短，由SRTN算法可知，先运行进程E。10时刻时，进程E执行完毕，进程B、D的剩余服务时间相同，因此先运行进程B。15时刻时，进程B执行完成，运行进程D，并在20时刻最终执行完所有作业。具体数据如下：



进程名	A	B	C	D	E	平均
完成时间	3	15	8	20	10	
周转时间	3	13	4	14	2	7.2
带权周转时间	1	2.17	1	2.8	1	1.59

1.4 高响应比优先(HRRN, Highest Response Ratio Next)

在0时刻时，只有进程A，因此首先运行进程A。进程A在3时刻执行完毕，此时只有进程B到达，因此运行进程B。9时刻时，进程B执行完毕，此时进程C、D、E皆已到达，其中C的优先权为1.83，D为1.6、E为1.5，进程C的优先权最大，先运行进程C。13时刻时，进程C执行完毕，D的优先权为2.4，E为3.5，因此先运行进程E。15时刻时，进程E执行完毕，运行进程D，并在20时刻最终执行完所有作业。具体数据如下：

进程名	A	B	C	D	E	平均
完成时间	3	9	13	20	15	
周转时间	3	7	9	14	7	8
带权周转时间	1	1.17	2.25	2.8	3.5	2.14

1.5 时间片轮转(RR, Round Robin)

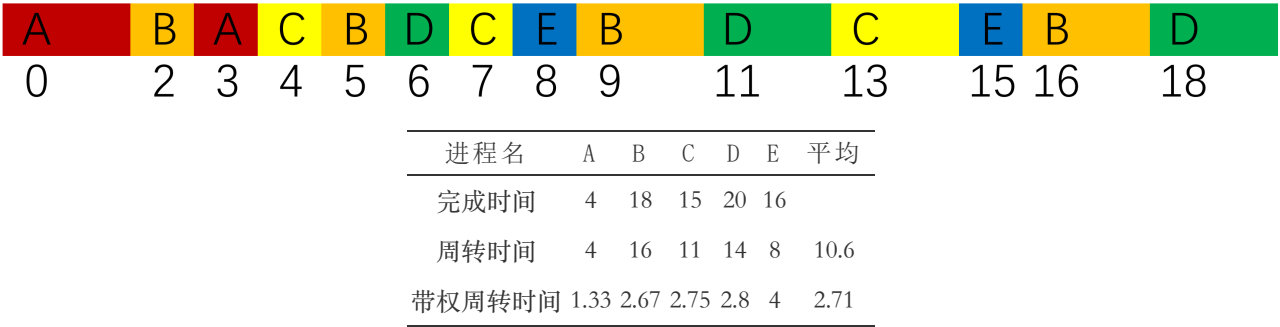
根据时间片轮转算法RR，此时时间片=1，且默认新到达的进程先进入就绪队列。在0时刻时，只有进程A(3)，因此首先运行进程A(3)。2时刻时，进程B(6)到达，进程B(6)先进入就绪队列，然后进程A(1)进入，先运行进程B(6)。3时刻时，进程B(5)下处理机入队列，运行进程A(1)。4时刻时，A执行完毕，进程C(4)到达，运行进程B(5)。5时刻时，B(4)下处理机入队列，运行进程C(4)。6时刻时，进程C(3)下处理机入队列，进程D(5)到达，运行进程B(4)。7时刻时，进程B(3)下处理机入队列，运行进程D(5)。8时刻时，进程D(4)下处理机入队列，进程E(2)到达，运行进程C(3)。9时刻时，进程C(2)下处理机入队列，运行进程B(3)，之后依次循环，具体过程如下图所示：



进程名	A	B	C	D	E	平均
完成时间	4	18	17	20	15	
周转时间	4	16	13	14	7	10.8
带权周转时间	1.33	2.67	3.25	2.8	3.5	2.71

1.6 多级反馈队列调度算法(Multi-level Feedback Queue, MLFQ)

具体流程略，如下图所示：



2 设在t时刻，系统中有5个进程，三种资源A、B、C的使用情况如下表所示，设系统可供使用的空闲资源数为Available=2，2，1。若此时有以下三种不同的分配请求，请用银行家算法判断系统能否予以分配，并写出具体步骤。(注：以下各小题没有因果关系，每小题都以表中所示状态为当前状态)

进程	Max			Need		
	A	B	C	A	B	C
P0	1	3	3	1	2	2
P1	3	2	2	2	2	1
P2	2	2	1	2	1	1
P3	2	0	2	1	0	1
P4	2	4	3	2	3	3

2.1 P3进程申请资源A、B、C分别为2、0、0，能否分配，为什么

不能。因为Request3=(2,0,0)，Need3=(1,0,1)，P3进程的资源申请量Request3[1]已经超过了它的需要值Need3[1]，即Request3[1]>Need3[1]，因此不予以分配。

2.2 P1进程申请资源A、B、C分别为0、0、2，能否分配，为什么

不能。因为Request1=(0,0,2)，Need1=(2,2,1)，P3进程的资源申请量Request1[3]已经超过了它的需要值Need1[3]，即Request1[3]>Need1[3]，因此不予以分配。

2.3 P2进程申请资源A、B、C分别为1、1、1，能否分配，为什么

能。Request2=(1,1,1)，Need2=(2,1,1)，Request2[1-3]<=Need2[1-3]。

系统先假定可为P2分配资源，修改相关向量值，如下所示：

资源 进程	Max			Allocation			Need			Available		
	A	B	C	A	B	C	A	B	C	A	B	C
P0	1	3	3	0	1	1	1	2	2	1	1	0
P1	3	2	2	1	0	1	2	2	1			
P2	2	2	1	1	2	1	1	0	0			
P3	2	0	2	1	0	1	1	0	1			
P4	2	4	3	0	1	0	2	3	3			

利用安全性算法检查此时系统是否安全。具体如表所示：

资源 进程	Work			Need			Allocation			Work+Allocation			Finish
	A	B	C	A	B	C	A	B	C	A	B	C	
P2	1	1	0	1	0	0	1	2	1	2	3	1	True
P0	2	3	1	1	2	2	0	1	1	2	4	2	True
P1	2	4	2	2	2	1	1	0	1	3	4	3	True
P3	3	4	3	1	0	1	1	0	1	4	5	4	True
P4	4	5	4	2	3	3	0	1	0	4	5	4	True

由此得得，可以找到一个安全序列{P2,P0,P1,P3,P4}（或{P2,P3,P1,P0,P4}等）。因此系统是安全的，可将P2所申请的资源分配给它。