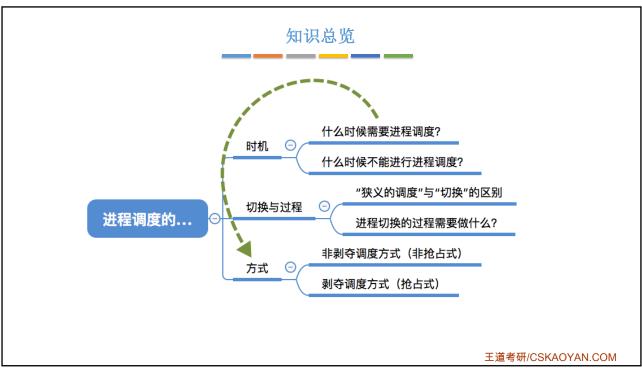


进程调度的 时机 切换与过程 调度方式

王道考研/CSKAOYAN.COM

1

公众号: 考研发条 一手课程!



进程调度的时机

进程调度(低级调度),就是按照某种算法从就绪队列中选择一个进程为其分配处理机。



当前运行的进 程主动放弃处 进程正常终止 运行过程中发生异常而终止 进程主动请求阻塞(如等待I/O)

WHEN?

需要进行进 程调度与切 换的情况



当前运行的进 程被动放弃处 理机

分给进程的时间片用完 有更紧急的事需要处理(如 I/O中断) 有更高优先级的进程进入就绪队列



- 1. 在处理中断的过程中。中断处理过程复杂,与硬件密切相关,很难 做到在中断处理过程中进行进程切换。
- 2. 进程在操作系统内核程序临界区中。
- 3. 在<mark>原子操作过程中</mark>(原语)。原子操作不可中断,要一气呵成(如 之前讲过的修改PCB中进程状态标志,并把PCB放到相应队列)

王道考研/CSKAOYAN.COM

3

公众号: 考研发条 一手课程!

进程调度的时机

进程在操作系统内核程序临界区中不能进行调度与切换



(2012年联考真题)进程处于临界区时不能进行处理机调度



临界资源:一个时间段内只允许一个进程使用的资源。各进程需要互斥地访问临界资源。 临界区:访问临界资源的那段代码。

内核程序临界区一般是用来访问某种内核数据结构的,比如进程的就绪队列(由各就绪进程的PCB组成)

就绪队列 内核程序临界区访问的临界资源 就进行 相关的 但此时 影响到操作系统内核的其他管理 工作。因此在访问内核程序临界 区期间不能进行调度与切换

普通临界区访问的临界资源不 会直接影响操作系统内核的管 理工作。因此在访问普通临界 区时可以进行调度与切换。

王道考研/CSKAOYAN.COM

进程调度的时机

进程调度(低级调度),就是按照某种算法从就绪队列中选择一个进程为其分配处理机。

有的系统中,只允许 进程主动放弃处理机

WHEN?

换的情况

需要进行进程调度与切

有的系统中,进程可以主动 放弃处理机,当有更紧急的 任务需要处理时,也会强行 剥夺处理机(被动放弃)





当前运行的进程主动放弃处 理机

进程正常终止 运行过程中发生异常而终止 进程主动请求阻塞(如 等待I/O)



当前运行的进程被动放弃处理机

分给进程的时间片用完 有更紧急的事需要处理(如 I/O中断) 有更高优先级的进程进入就绪队列

但是进程在普通临界区中是

- 1. 在处理中断的过程中。中断处理过一可以进行调度、切换的。 做到在中断处理过程中进行进程切换。
- 2. 进程在操作系统内核程序临界区中。
- 3. 在原子操作过程中(原语)。原子操作不可中断,要一气呵成(如 之前讲过的修改PCB中程序状态标志,并把PCB放到相应队列)

王道考研/CSKAOYAN.COM

5

公众号: 考研发条 一手课程!

进程调度的方式



<mark>非剥夺调度方式</mark>,又称<mark>非抢占方式</mark>。即,只允许进程主动放弃处理机。在运行过程中即便有更紧迫 的任务到达,当前进程依然会继续使用处理机,直到该进程终止或主动要求进入阻塞态。

实现简单,系统开销小但是无法及时处 理紧急任务,适合于早期的批处理系统



<mark>剥夺调度方式</mark>,又称<mark>抢占方式</mark>。当一个进程正在处理机上执行时,如果有一个更重要或更紧迫的进程需要使用处理机,则立即暂停正在执行的进程,将处理机分配给更重要紧迫的那个进程。

可以优先处理更紧急的进程,也可实现让各 进程按时间片轮流执行的功能(通过时钟中 断)。适合于分时操作系统、实时操作系统

王道考研/CSKAOYAN.COM

进程的切换与过程

"狭义的进程调度"与"进程切换"的区别:

狭义的进程调度指的是从就绪队列中选中一个要运行的进程。(这个进程可以是刚刚被暂停执行的进程,也可能是另一个进程,后一种情况就需要进程切换)

进程切换是指一个进程让出处理机,由另一个进程占用处理机的过程。

广义的进程调度包含了选择一个进程和进程切换两个步骤。

进程切换的过程主要完成了:

- 1. 对原来运行进程各种数据的保存
- 2. 对新的进程各种数据的恢复

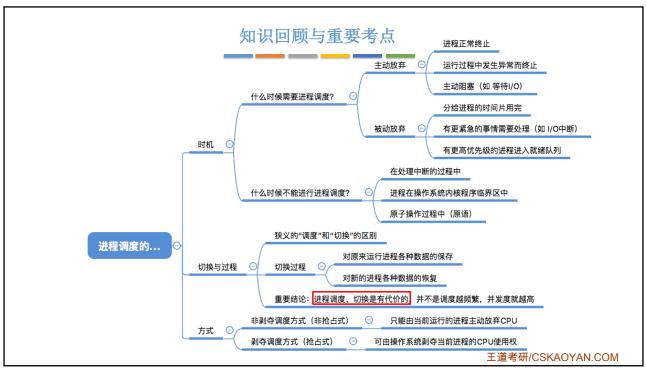
(如:程序计数器、程序状态字、各种数据寄存器等处理机现场信息,这些信息一般保存在进程控制块)

注意: 进程切换是有代价的,因此如果过于频繁的进行进程调度、切换,必然会使整个系统的效率降低,使系统大部分时间都花在了进程切换上,而真正用于执行进程的时间减少。

王道考研/CSKAOYAN.COM

7

公众号: 考研发条 一手课程!





9

公众号: 考研发条 一手课程!