

# 进程 及其实现

Linux  
Android  
Linux  
OpenStack  
Mac OS  
Windows



# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素

# 进程定义



## 定义

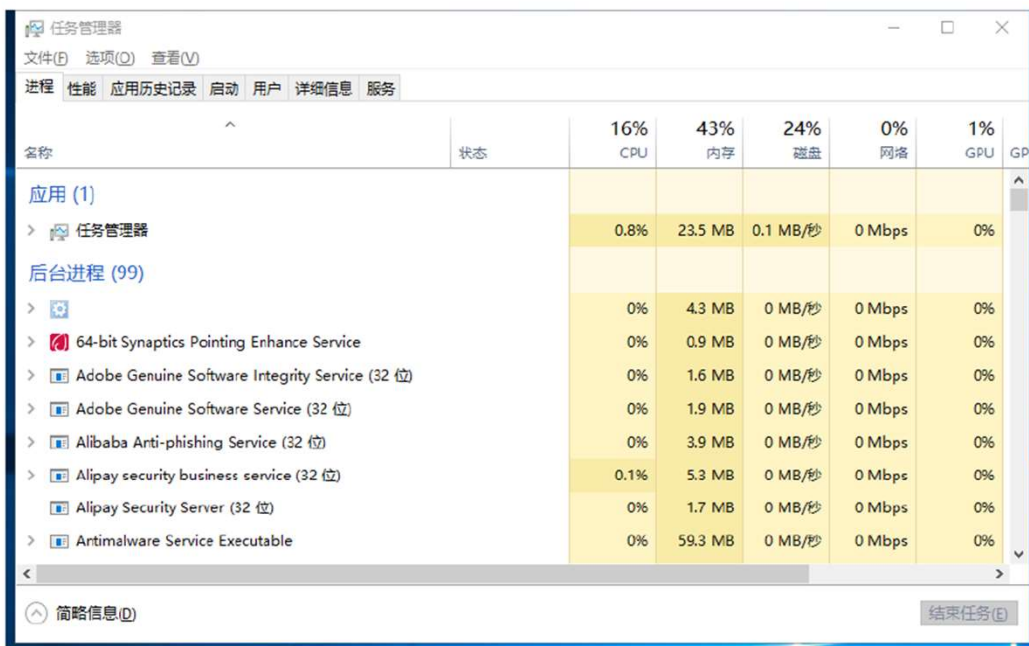
- ❏ 进程是为了描述程序在并发执行时对系统资源的共享，所需的一个描述程序执行时动态特征的概念。
- ❏ 进程是具有独立功能的程序关于某个数据集合上的一次运行活动，是系统进行资源分配、调度和保护的一个独立单位。

# 进程的类型和特性

## 3 程序与进程之间的区别

- ❏ 进程能真实地描述并发，而程序不能
- ❏ 进程是由程序和数据和控制块组成
- ❏ 程序是静态的，进程是动态的
- ❏ 进程有生命周期，程序是相对长久的
- ❏ 一个程序可对应多个进程，反之亦然
- ❏ 进程具有创建其它进程的功能，程序没有

# 进程的概念



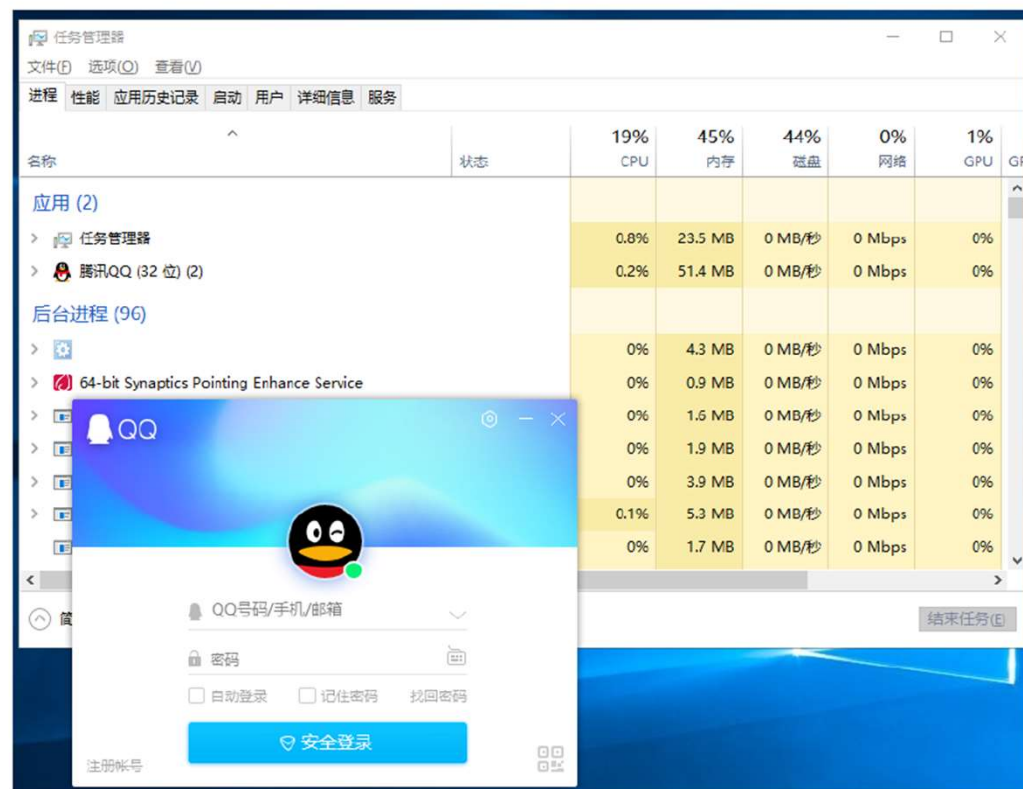
任务管理器

文件(F) 选项(O) 查看(V)

进程 性能 应用历史记录 启动 用户 详细信息 服务

名称	状态	16% CPU	43% 内存	24% 磁盘	0% 网络	1% GPU	GP
<b>应用 (1)</b>							
> 任务管理器		0.8%	23.5 MB	0.1 MB/秒	0 Mbps	0%	
<b>后台进程 (99)</b>							
> 64-bit Synaptics Pointing Enhance Service		0%	4.3 MB	0 MB/秒	0 Mbps	0%	
> Adobe Genuine Software Integrity Service (32 位)		0%	0.9 MB	0 MB/秒	0 Mbps	0%	
> Adobe Genuine Software Service (32 位)		0%	1.6 MB	0 MB/秒	0 Mbps	0%	
> Alibaba Anti-phishing Service (32 位)		0%	3.9 MB	0 MB/秒	0 Mbps	0%	
> Alipay security business service (32 位)		0.1%	5.3 MB	0 MB/秒	0 Mbps	0%	
> Alipay Security Server (32 位)		0%	1.7 MB	0 MB/秒	0 Mbps	0%	
> Antimalware Service Executable		0%	59.3 MB	0 MB/秒	0 Mbps	0%	

任务管理器



任务管理器

文件(F) 选项(O) 查看(V)

进程 性能 应用历史记录 启动 用户 详细信息 服务

名称	状态	19% CPU	45% 内存	44% 磁盘	0% 网络	1% GPU	GP
<b>应用 (2)</b>							
> 任务管理器		0.8%	23.5 MB	0 MB/秒	0 Mbps	0%	
> 腾讯QQ (32 位) (2)		0.2%	51.4 MB	0 MB/秒	0 Mbps	0%	
<b>后台进程 (96)</b>							
> 64-bit Synaptics Pointing Enhance Service		0%	4.3 MB	0 MB/秒	0 Mbps	0%	
> Adobe Genuine Software Integrity Service (32 位)		0%	0.9 MB	0 MB/秒	0 Mbps	0%	
> Adobe Genuine Software Service (32 位)		0%	1.6 MB	0 MB/秒	0 Mbps	0%	
> Alibaba Anti-phishing Service (32 位)		0%	1.9 MB	0 MB/秒	0 Mbps	0%	
> Alipay security business service (32 位)		0%	3.9 MB	0 MB/秒	0 Mbps	0%	
> Alipay Security Server (32 位)		0.1%	5.3 MB	0 MB/秒	0 Mbps	0%	
> Antimalware Service Executable		0%	1.7 MB	0 MB/秒	0 Mbps	0%	

任务管理器

QQ 登录对话框

QQ 号码/手机/邮箱

密码

自动登录  记住密码 找回密码

安全登录

注册帐号

# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素

# 进程的类型和特性

## ① 进程分类

系统进程

用户进程

# 进程的类型和特性

## 2 进程的属性

结构性

共享性

动态性

独立性

制约性

并发性

## 进程的特征

程序是静态的，进程是动态的，相比于程序，进程拥有以下特征：

动态性是进程最基本的特征

### 进程的特征

- 动态性 ⊖ 进程是程序的一次执行过程，是动态地产生、变化和消亡的
- 并发性 ⊖ 内存中有多个进程实体，各进程可并发执行
- 独立性 ⊖ 进程是能独立运行、独立获得资源、独立接受调度的基本单位
- 异步性 ⊖ 各进程按各自独立的、不可预知的速度向前推进，操作系统要提供“进程同步机制”来解决异步问题
- 结构性 ⊖ 每个进程都会配置一个PCB。结构上看，进程由程序段、数据段、PCB组成

异步性会导致并发程序执行结果的不确定性。具体会在“进程同步”相关小节进行学习

# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素

# 进程的状态和转换

## 1 三种进程状态



**运行态 (Running) :**

- ❏ 进程占有CPU，并在CPU上运行



**就绪态 (Ready) :**

- ❏ 一个进程已经具备运行条件，但没有分配CPU，暂时不能运行
- ❏ 当调度给该进程CPU时，立即可以运行

# 进程的状态和转换

## 1 三种进程状态



### 等待态 (Blocked) :

- ❏ 阻塞态、封锁态、睡眠态
- ❏ 进程因等待某事件的发生而暂时不能运行的状态
- ❏ 即使CPU空闲，该进程也不可运行

## 知识滚雪球：程序是如何运行的？



C语言  
代码

编译器  
“翻译”

机器指令  
(二进制)

一条高级语言的代码翻译过来可能会对应多条机器指令

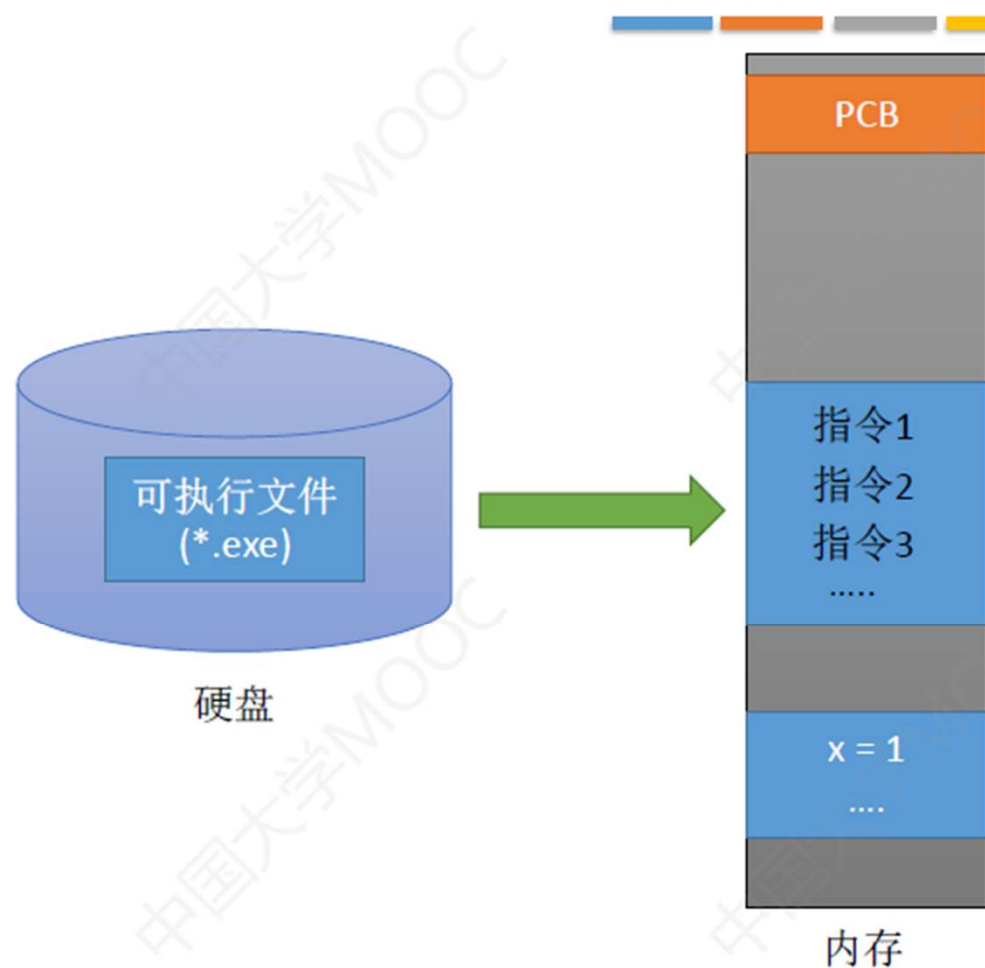
```
int x = 1;  
x++;  
.....
```

指令1  
指令2  
指令3  
.....



程序运行的过程其实就是CPU执行一条一条的机器指令的过程

## 进程的状态——创建态、就绪态

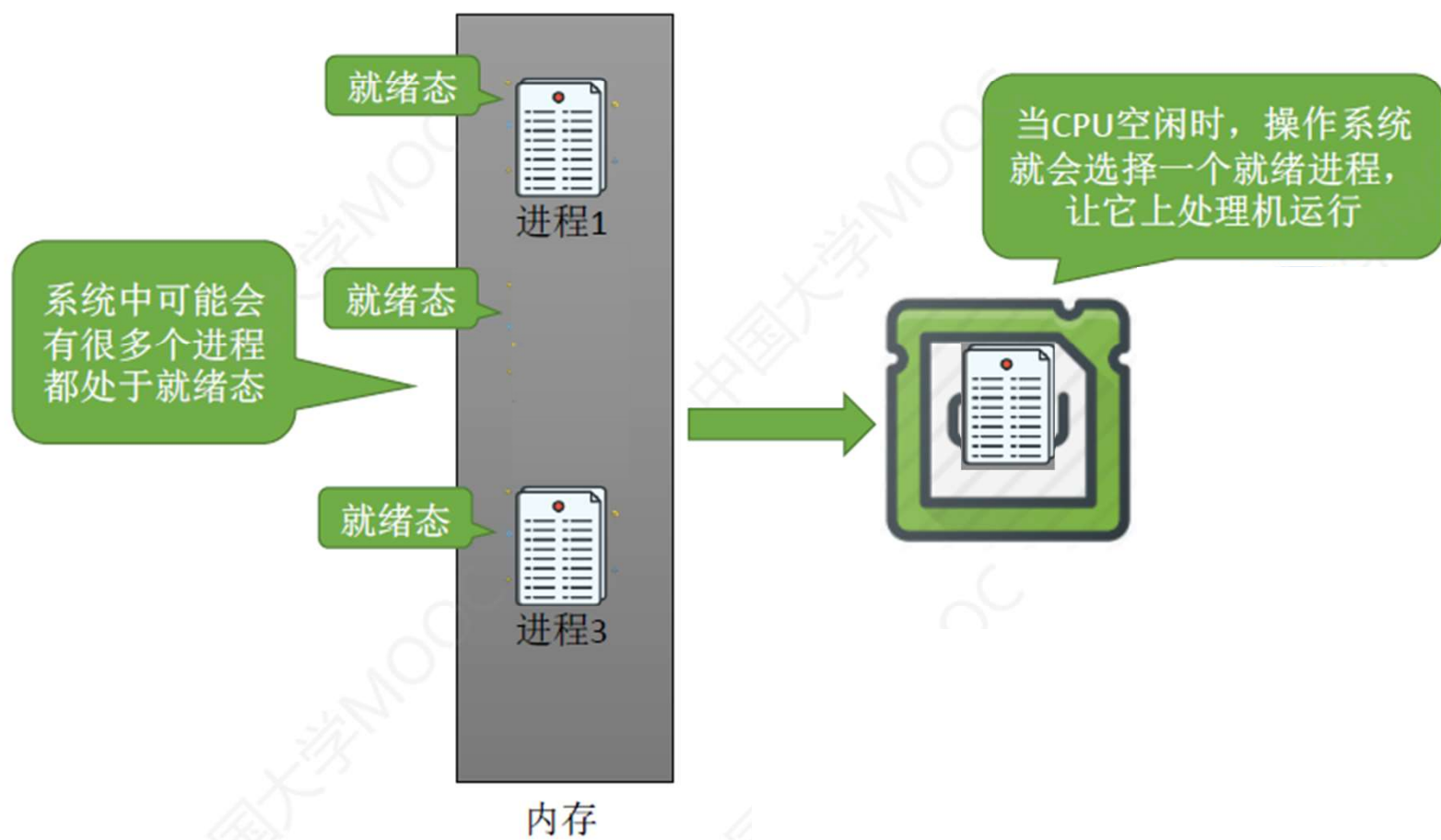


进程正在被创建时，它的状态是“**创建态**”，在这个阶段操作系统会为进程分配资源、初始化PCB

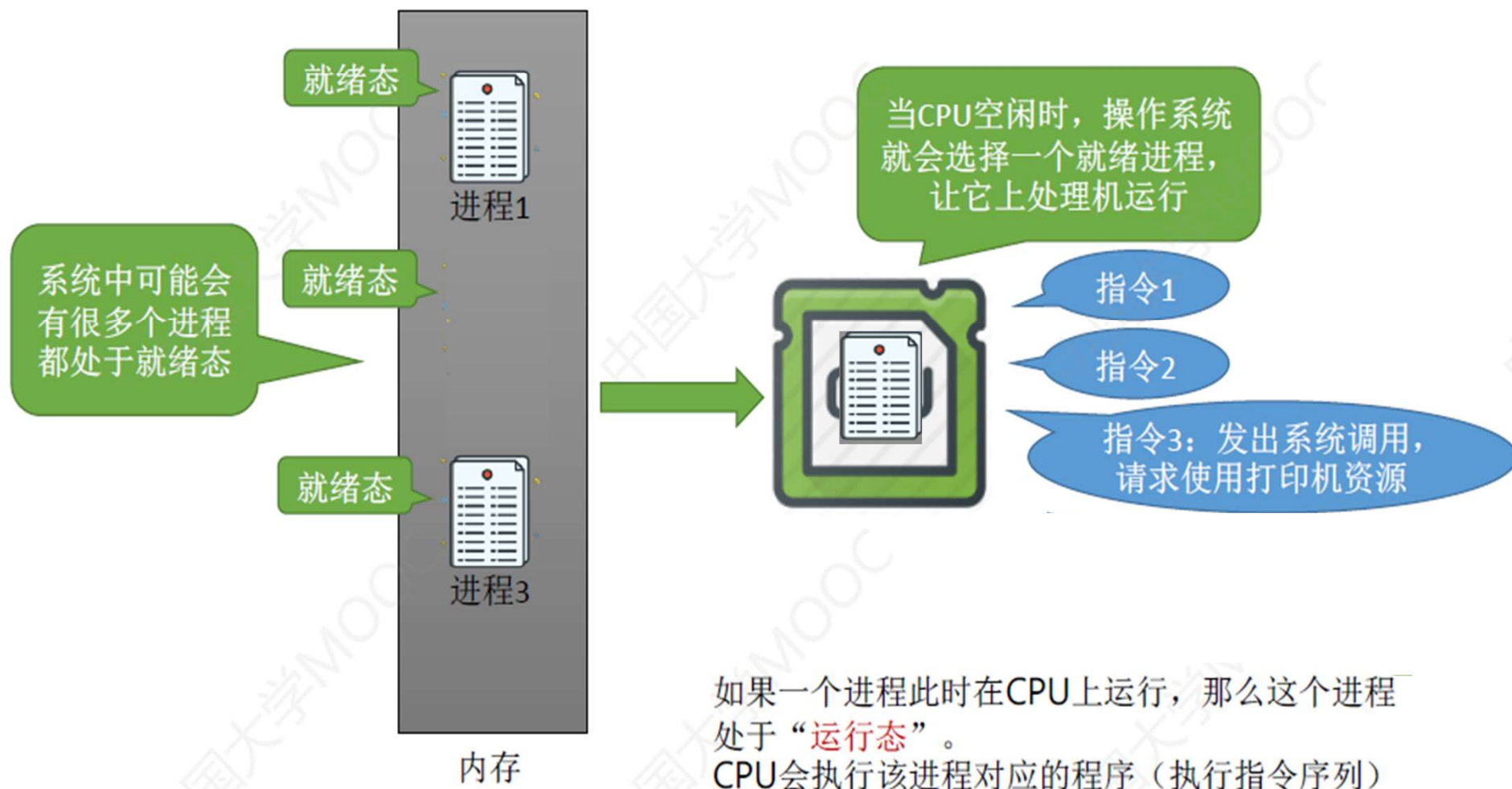


当进程创建完成后，便进入“**就绪态**”，处于就绪态的进程已经具备运行条件，但由于没有空闲CPU，就暂时不能运行

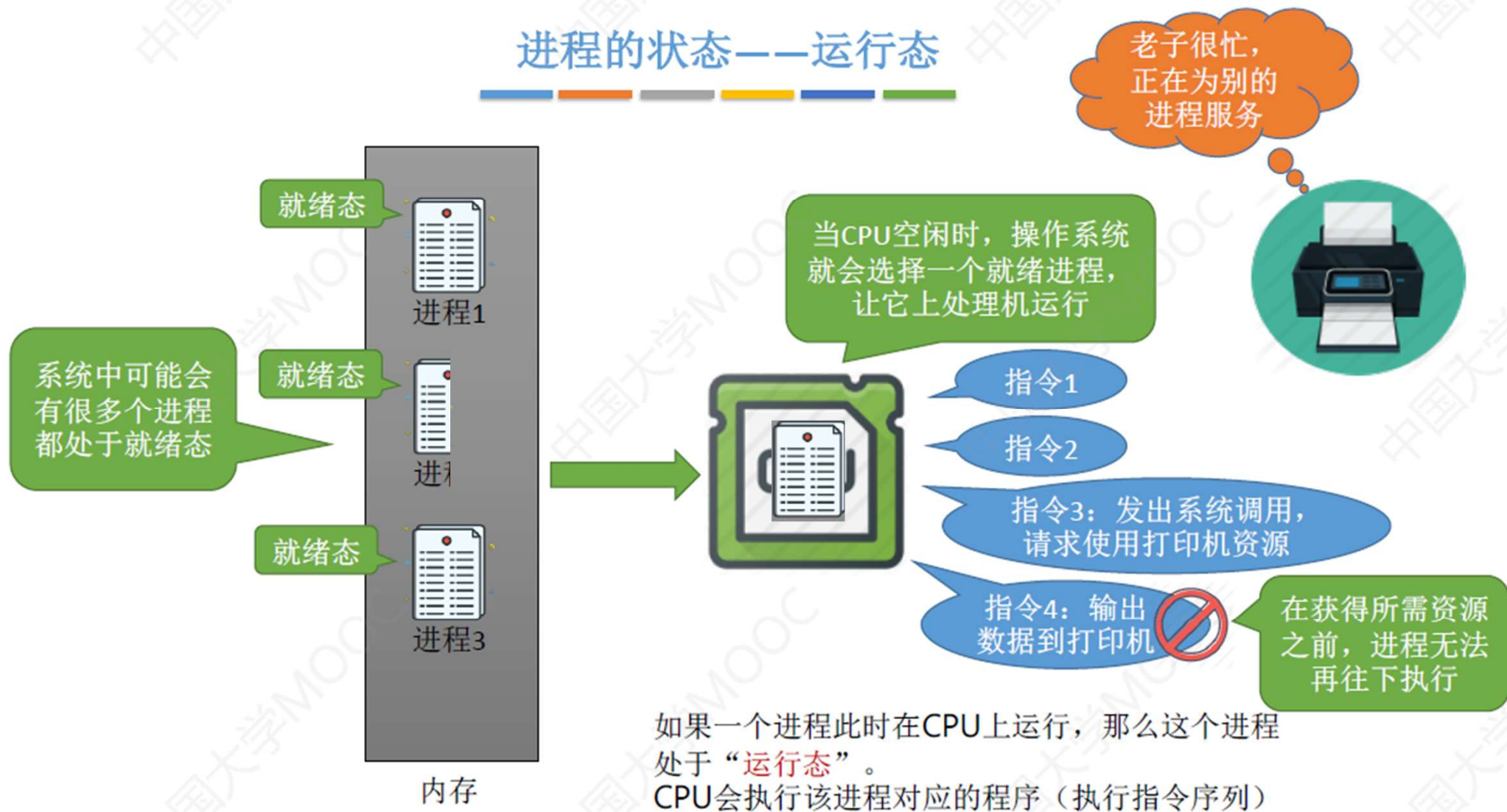
## 进程的状态——运行态



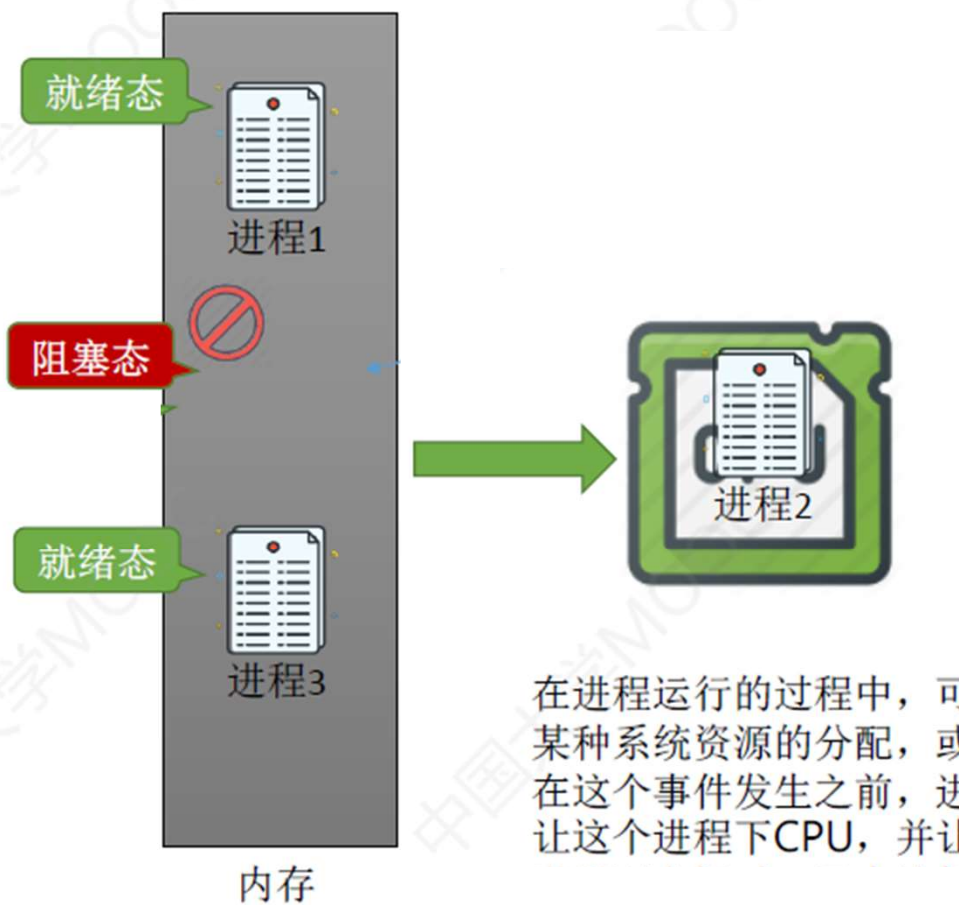
## 进程的状态——运行态



## 进程的状态——运行态



## 进程的状态——阻塞态



老子很忙，  
正在为别的  
进程服务



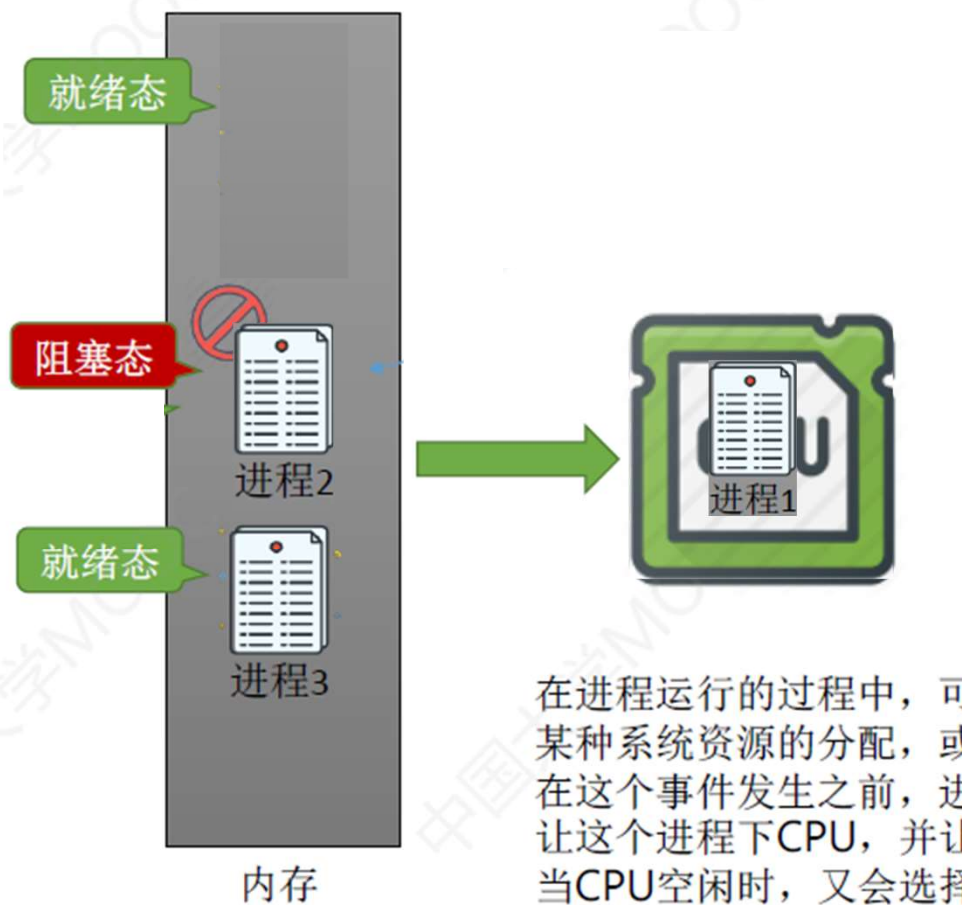
在进程运行的过程中，可能会请求等待某个事件的发生（如等待某种系统资源的分配，或者等待其他进程的响应）。在这个事件发生之前，进程无法继续往下执行，此时操作系统会让这个进程下CPU，并让它进入“阻塞态”

## 进程的状态——阻塞态



在进程运行的过程中，可能会请求等待某个事件的发生（如等待某种系统资源的分配，或者等待其他进程的响应）。在这个事件发生之前，进程无法继续往下执行，此时操作系统会让这个进程下CPU，并让它进入“阻塞态”

## 进程的状态——阻塞态

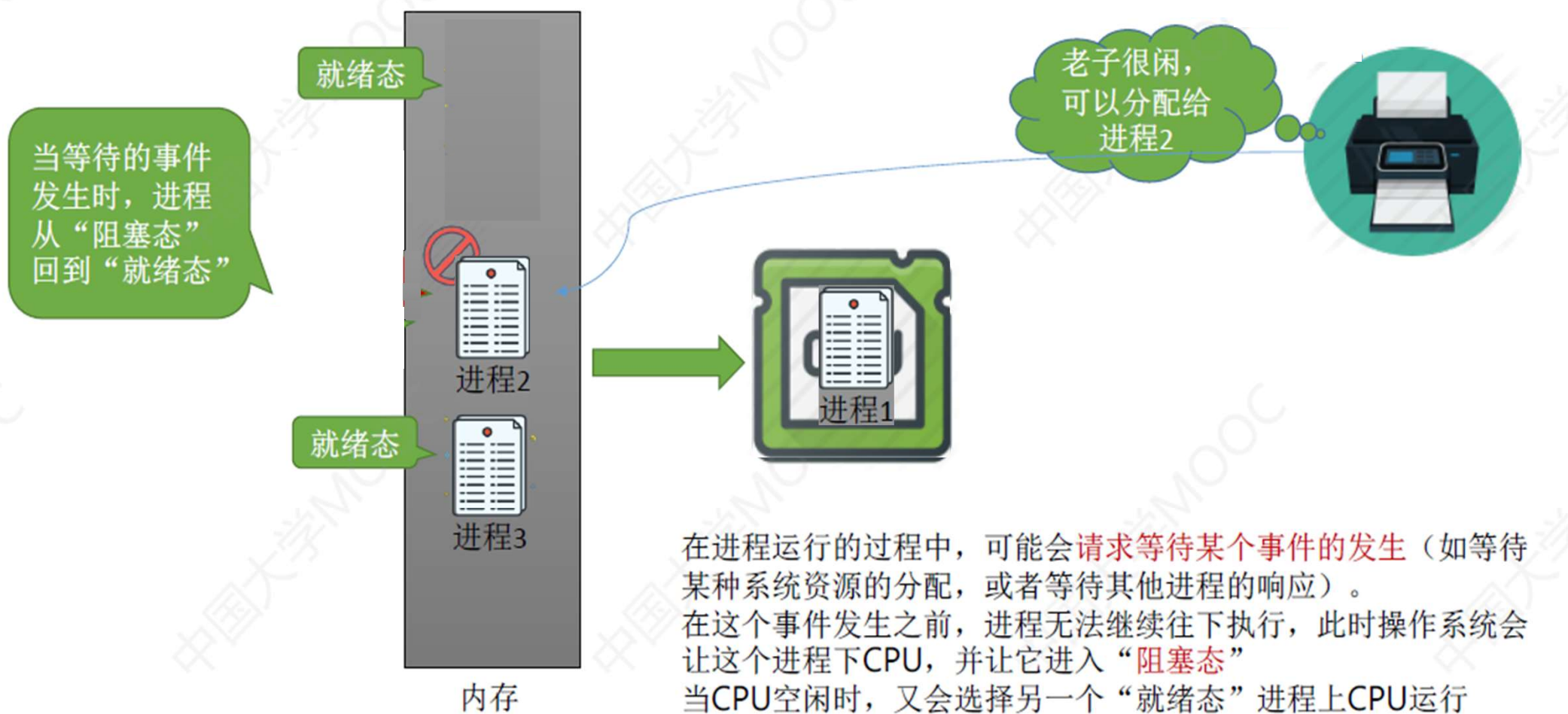


老子很忙，  
正在为别的  
进程服务

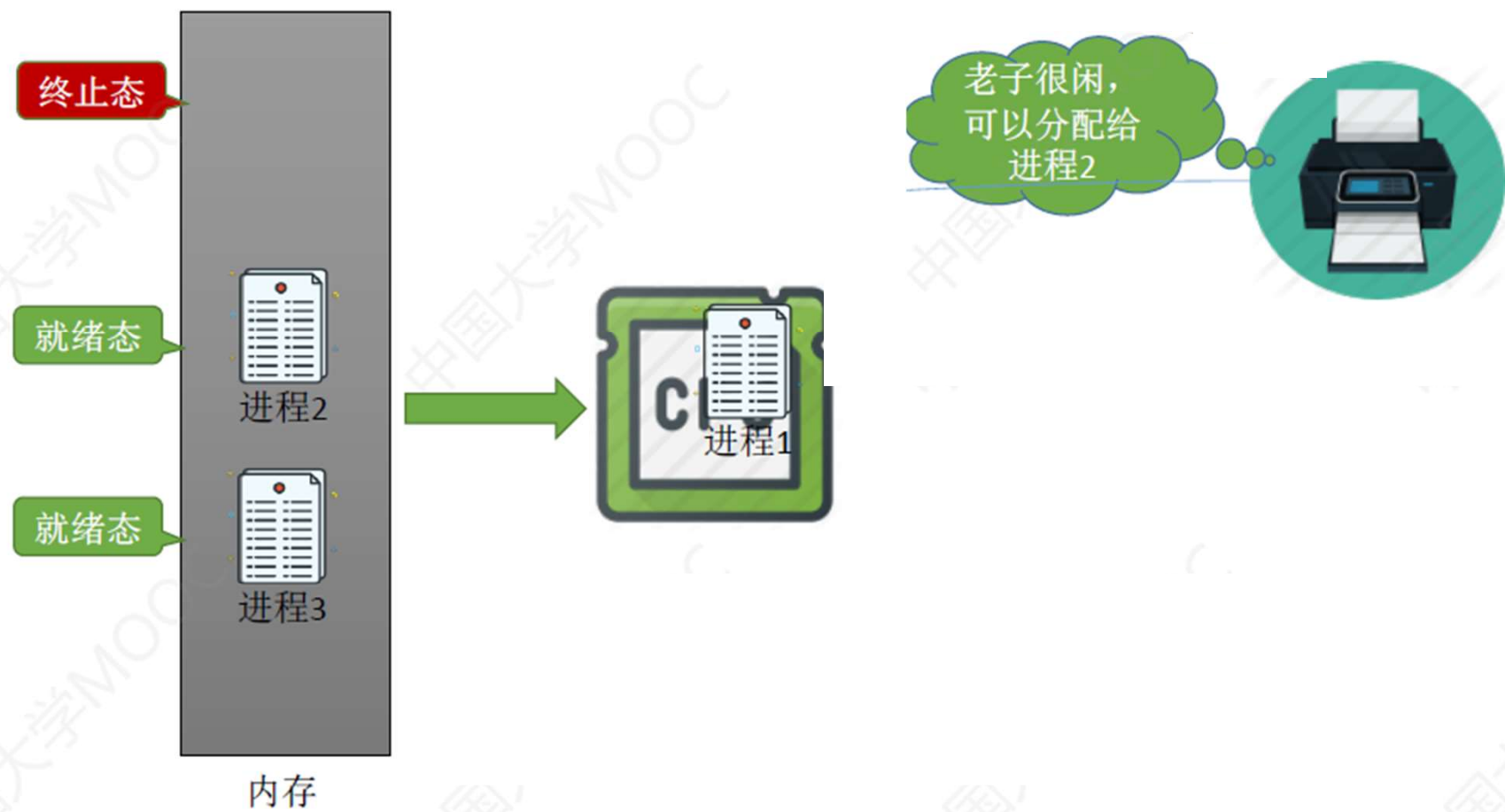


在进程运行的过程中，可能会请求等待某个事件的发生（如等待某种系统资源的分配，或者等待其他进程的响应）。在这个事件发生之前，进程无法继续往下执行，此时操作系统会让这个进程下CPU，并让它进入“阻塞态”。当CPU空闲时，又会选择另一个“就绪态”进程上CPU运行。

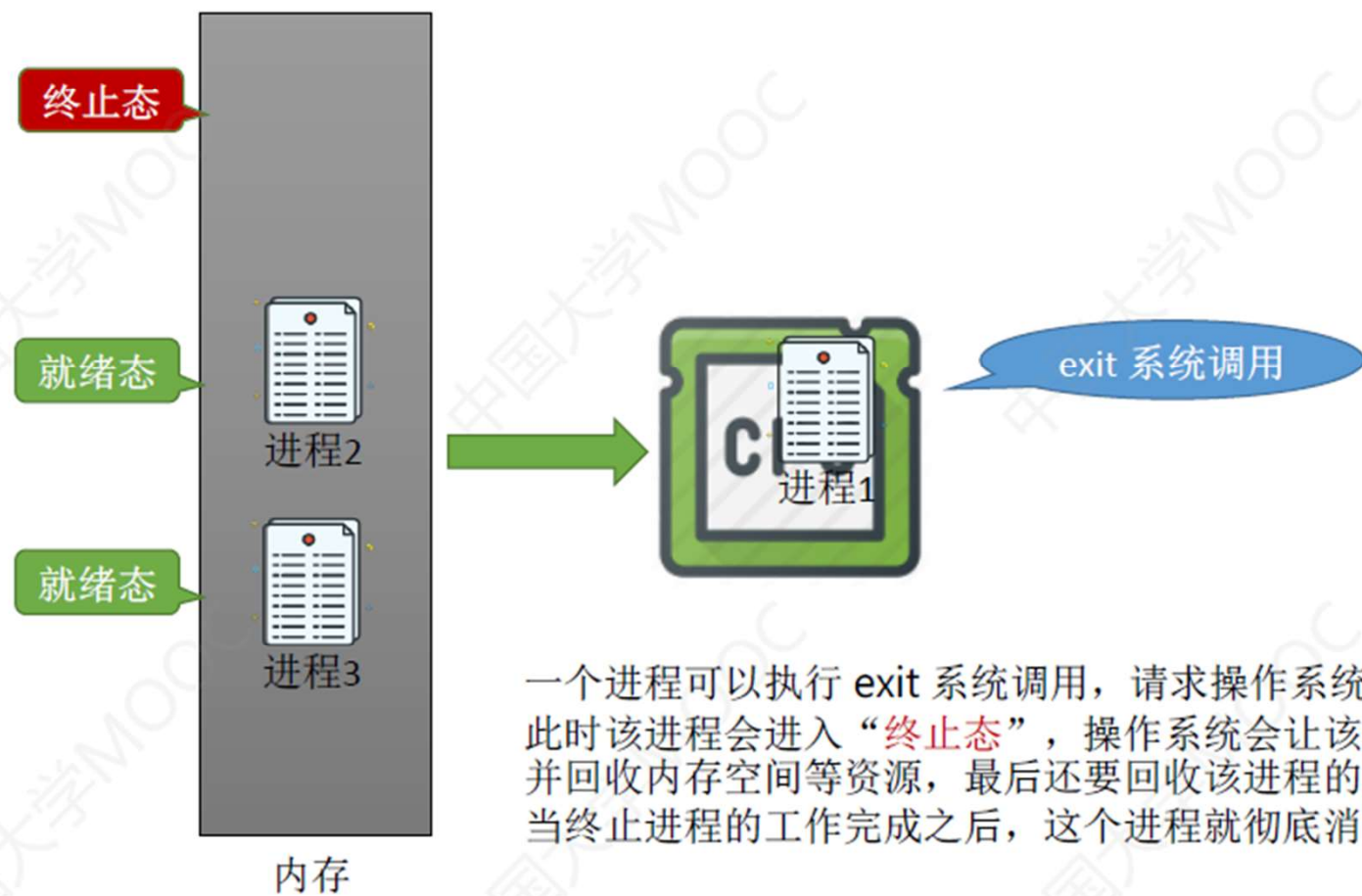
## 进程的状态——阻塞态



## 进程的状态——终止态



## 进程的状态——终止态



一个进程可以执行 `exit` 系统调用，请求操作系统终止该进程。此时该进程会进入“**终止态**”，操作系统会让该进程下CPU，并回收内存空间等资源，最后还要回收该进程的PCB。当终止进程的工作完成之后，这个进程就彻底消失了。

# 进程的状态和转换

## 1 三种进程状态

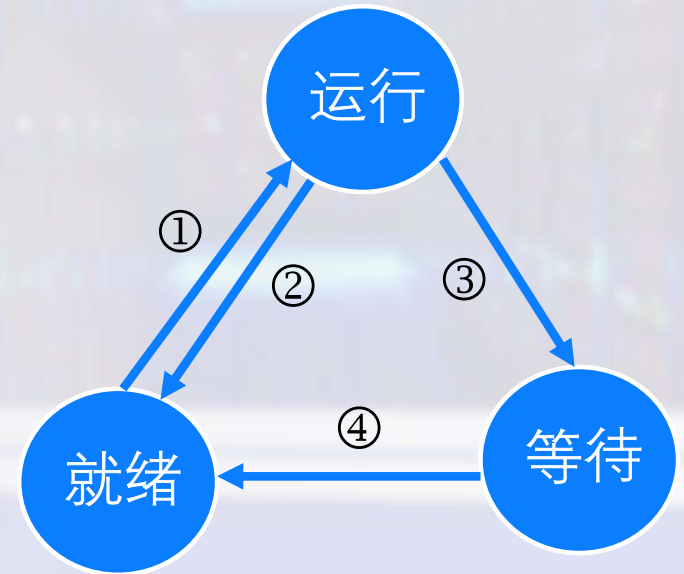
就绪 --> 运行

调度程序选择进程运行

运行 --> 就绪

运行进程用完了时间片

当时间片到或者处理机被抢占了，就转换到就绪态；



# 进程的状态和转换

## 1 三种进程状态

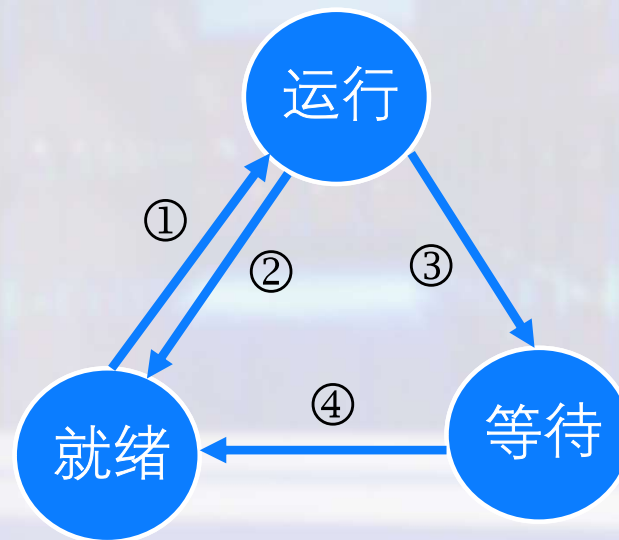
运行 --> 等待

操作系统尚未完成服务

对资源的访问尚不能进行

初始化I/O且须等待结果

等待某一进程提供输入



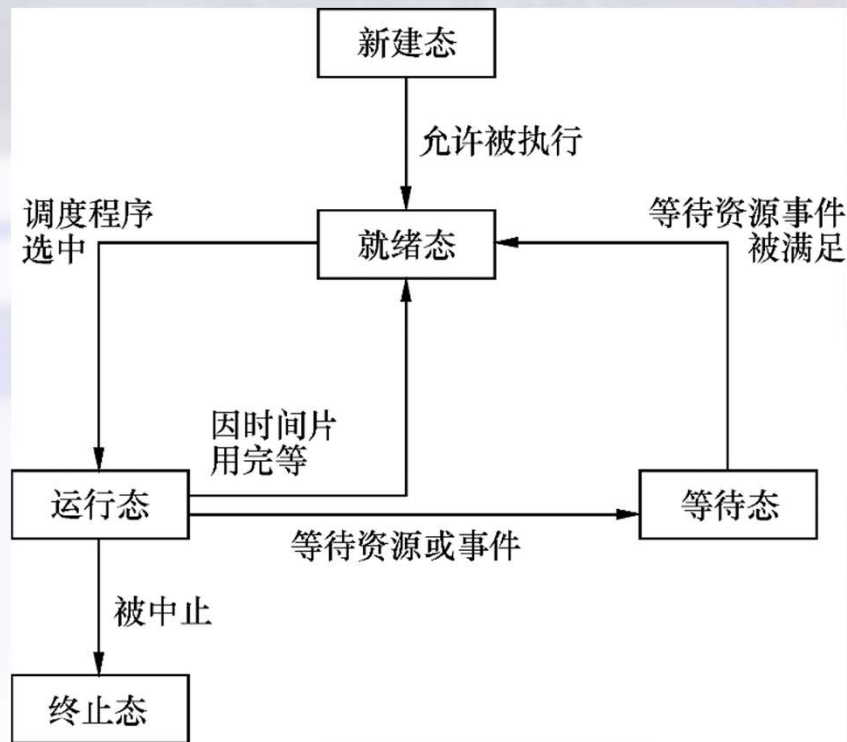
等待 --> 就绪

等待的事件发生

# 进程的状态和转换

## 2 五种进程状态

三种进程状态 → 五状态进程模型



# 进程的状态和转换

## 2 五种进程状态

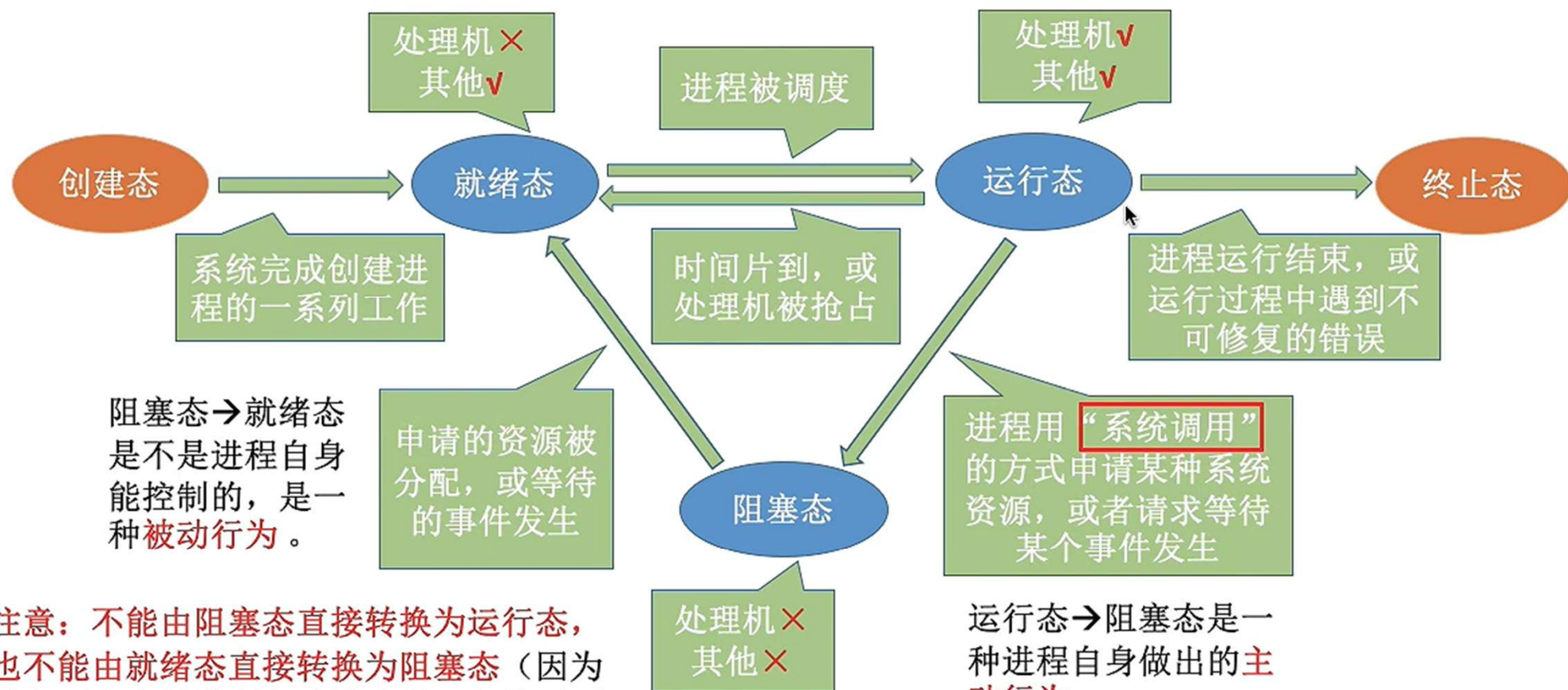
- ❏ 无→新建态：创建子进程
- ❏ 新建态→就绪态：系统完成进程创建操作，且当前系统的性能和内存的容量均允许
- ❏ 运行态→终止态：进程到达自然结束点，或出现了无法克服的错误，或被操作系统所终结，或被其它有终止权的进程所终结

# 进程的状态和转换

## 2 五种进程状态

- ❏ 终止态→NULL: 完成善后操作
- ❏ 就绪态→终止态: 父进程终结子进程
- ❏ 等待态→终止态: 父进程终结子进程

## 进程状态的转换



注意：不能由阻塞态直接转换为运行态，也不能由就绪态直接转换为阻塞态（因为进入阻塞态是进程主动请求的，必然需要进程在运行时才能发出这种请求）

运行态→阻塞态是一种进程自身做出的主动行为

## 进程的状态

进程的整个生命周期中，大部分时间都处于三种基本状态

单CPU情况下，同一时刻只会会有一个进程处于运行态，多核CPU情况下，可能有多个进程处于运行态

### 三种基本状态

运行态 (Running)

占有CPU，并在CPU上运行

就绪态 (Ready)

已经具备运行条件，但由于没有空闲CPU，而暂时不能运行

阻塞态 (Waiting/Blocked, 又称：等待态)

因等待某一事件而暂时不能运行

### 另外两种状态

创建态 (New, 又称：新建态)

进程正在被创建，操作系统为进程分配资源、初始化PCB

终止态 (Terminated, 又称：结束态)

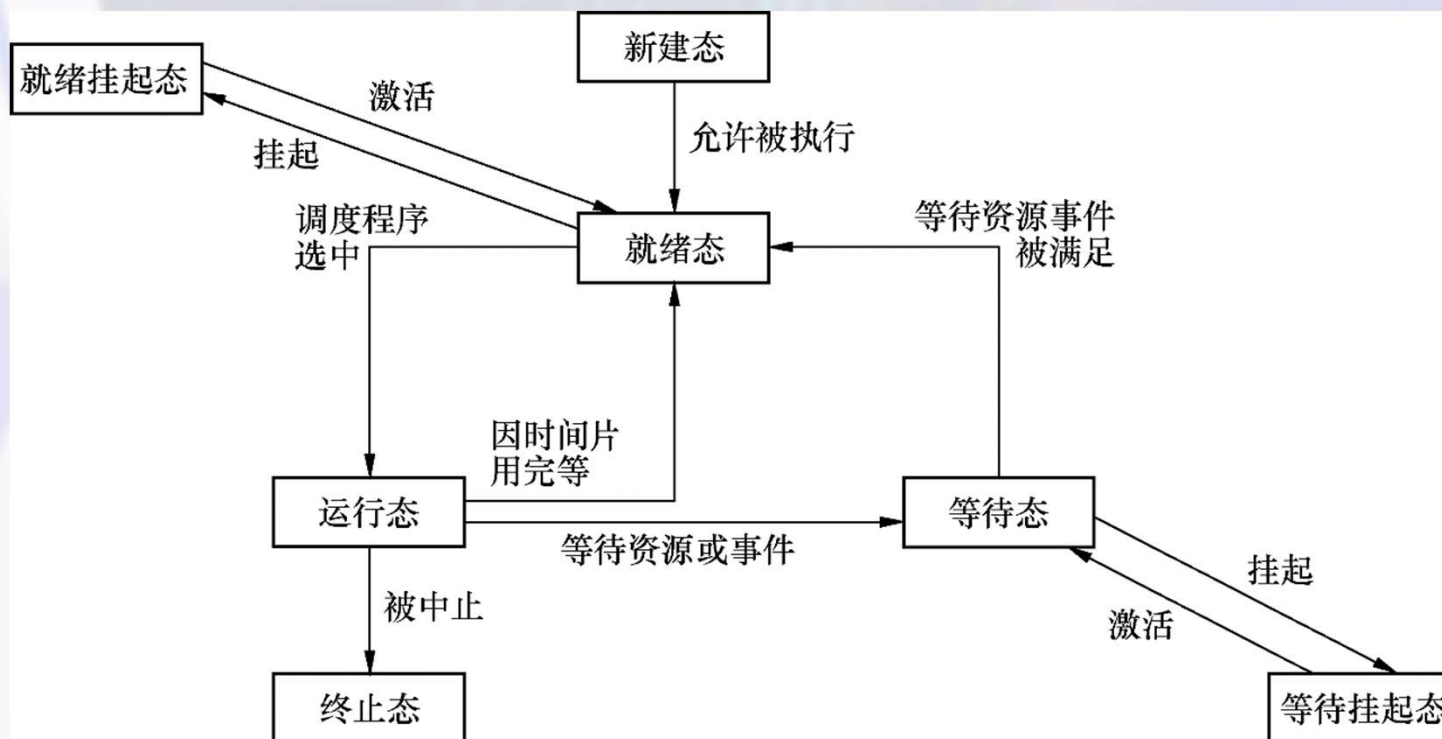
进程正在从系统中撤销，操作系统会回收进程拥有的资源、撤销PCB

进程PCB中，会有一个变量 `state` 来表示进程的当前状态。如：1表示创建态、2表示就绪态、3表示运行态... 为了对同一个状态下的各个进程进行统一的管理，操作系统会将各个进程的PCB组织起来。

# 进程的状态和转换

## 3 七种进程状态

五状态进程模型 → 七状态进程模型



# 进程的状态和转换

## 3 七种进程状态

五状态进程模型 → 七状态进程模型



为什么要有“挂起”状态？

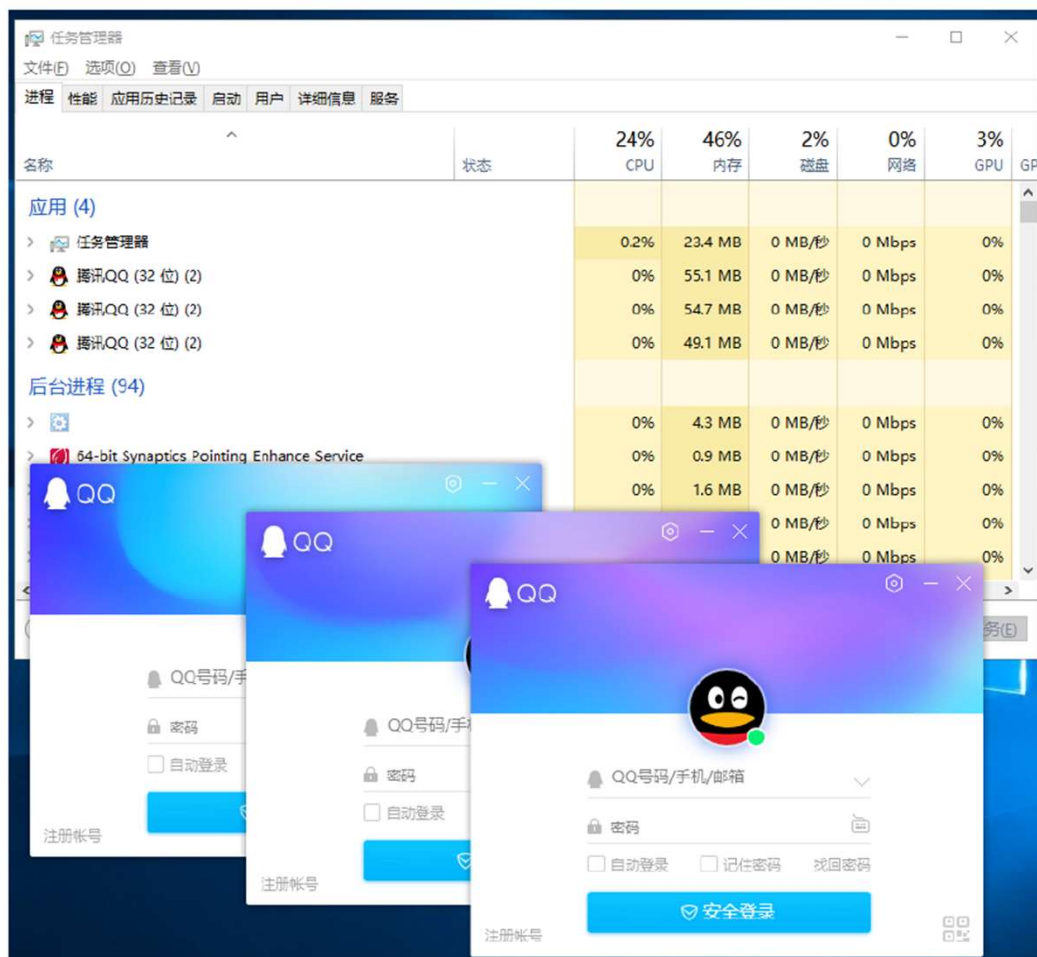
- ❏ 进程的不断创建，系统资源(内存)已不能满足进程运行的要求
- ❏ 某些进程挂起，对换到磁盘镜像区中，暂时不参与进程调度，平滑系统操作的负荷

# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素

## 进程的概念



程序：是**静态的**，就是个存放在磁盘里的可执行文件，就是一系列的指令集合。

进程（Process）：是**动态的**，是程序的一次执行过程

同一个程序多次执行会对应多个进程

思考：操作系统是这些进程的管理者，它要怎么区分各个进程？



# 进程控制块

## 1 定义

❖ 进程控制块（Process Control Block, PCB）是系统为了管理进程设置的专门数据结构，用来记录进程的外部特征，描述进程的变化过程

❖ 系统利用PCB来控制和管理进程，PCB是系统感知进程存在的唯一标志

❖ 进程与PCB一一对应

PCB 是系统感知进程的唯一方式  
删除某个进程的PCB，等于在系统中清除该进程

# 进程控制块

## 2 构成

进程描述信息

进程控制信息

所拥有的资源和使用情况

CPU 现场保护信息

# 进程控制块

## 2 构成

### 1. 进程描述信息

- ❏ 进程标识符(process ID), 唯一, 通常是一个整数
- ❏ 进程名, 通常基于可执行文件名 (不唯一)
- ❏ 用户标识符(user ID)
- ❏ 进程组关系

# 进程控制块

## 2 构成

### 2. 进程控制信息

- ▣ 当前状态
- ▣ 优先级
- ▣ 代码执行入口地址
- ▣ 程序的外存地址
- ▣ 运行统计信息
- ▣ 进程间同步和通信
- ▣ 阻塞原因
- ▣ 进程的队列指针
- ▣ 进程的消息队列指针

# 进程控制块

## 2 构成

### 3、所拥有的资源和使用情况

虚拟地址空间的现状

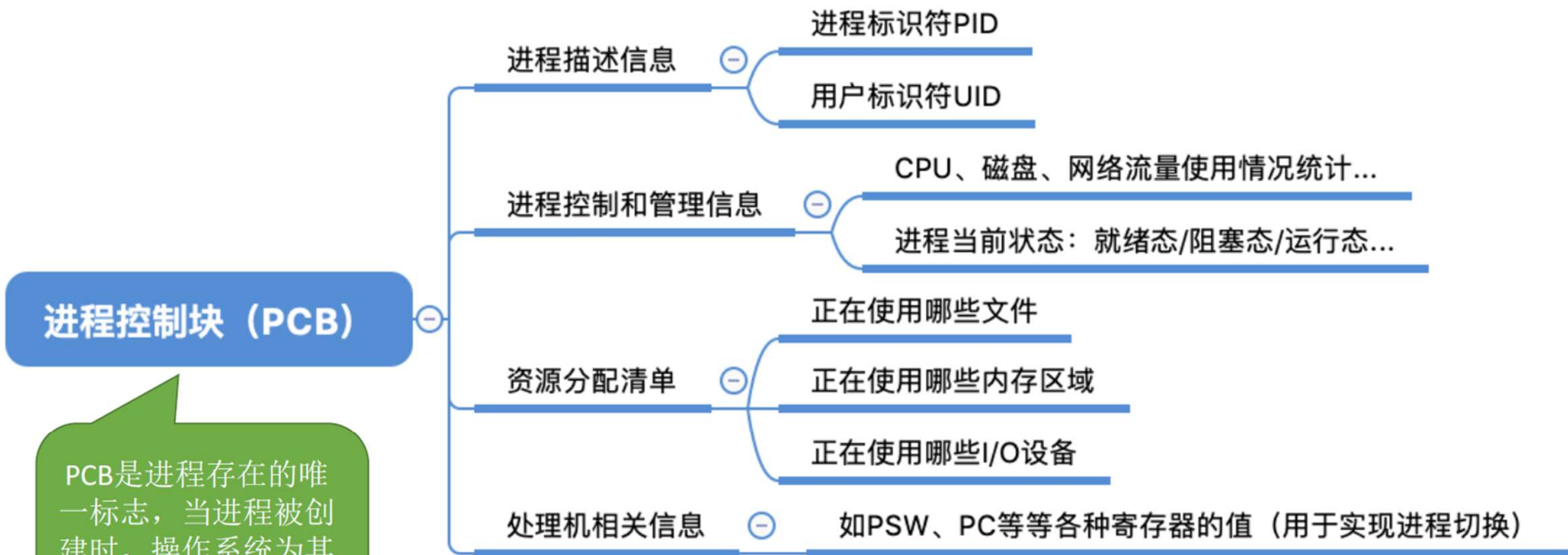
打开文件列表

### 4、CPU现场保护信息

寄存器值（通用、程序计数器PC、状态PSW，地址包括栈指针）

指向赋予该进程的段/页表的指针

## 进程的组成——PCB



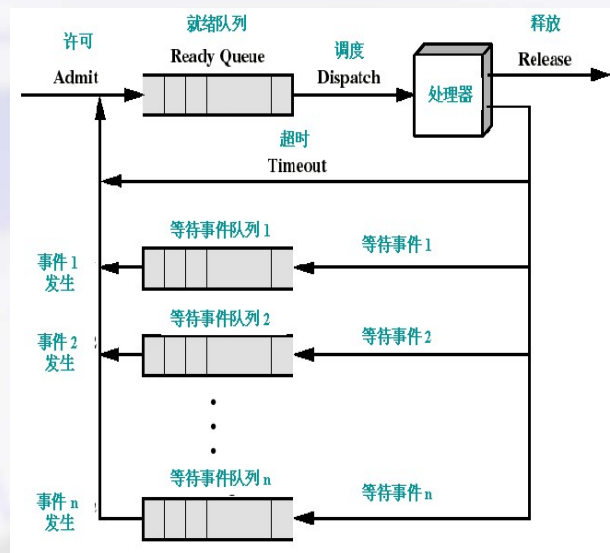
PCB是进程存在的唯一标志，当进程被创建时，操作系统为其创建PCB，当进程结束时，会回收其PCB。

操作系统对进程进行管理所需的信息都存在PCB中

# 进程控制块

## 3 组织

- 系统把PCB组织在一起，放在内存，构成PCB表
- PCB表大小决定了系统中最多可同时存在的进程个数，称为系统的并发度



1. 就绪队列

2. 等待队列

就绪状态的进程

阻塞状态的进程

- 等待队列为多个
- 等待同一类资源的放在同一个队列中

# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素

# 进程要素

## 1 构成

PCB是给操作系统用

进程控制块

程序段、数据段是给进程自己用

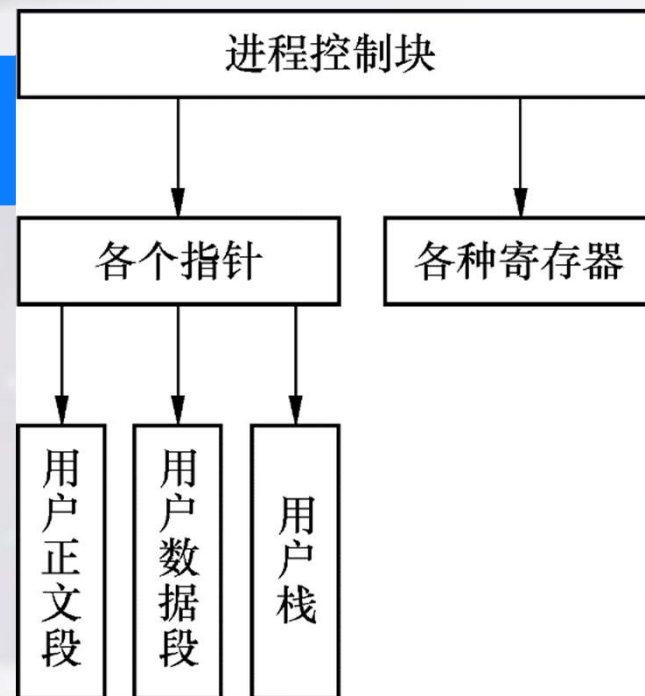
进程程序

: 程序的代码(指令序列)

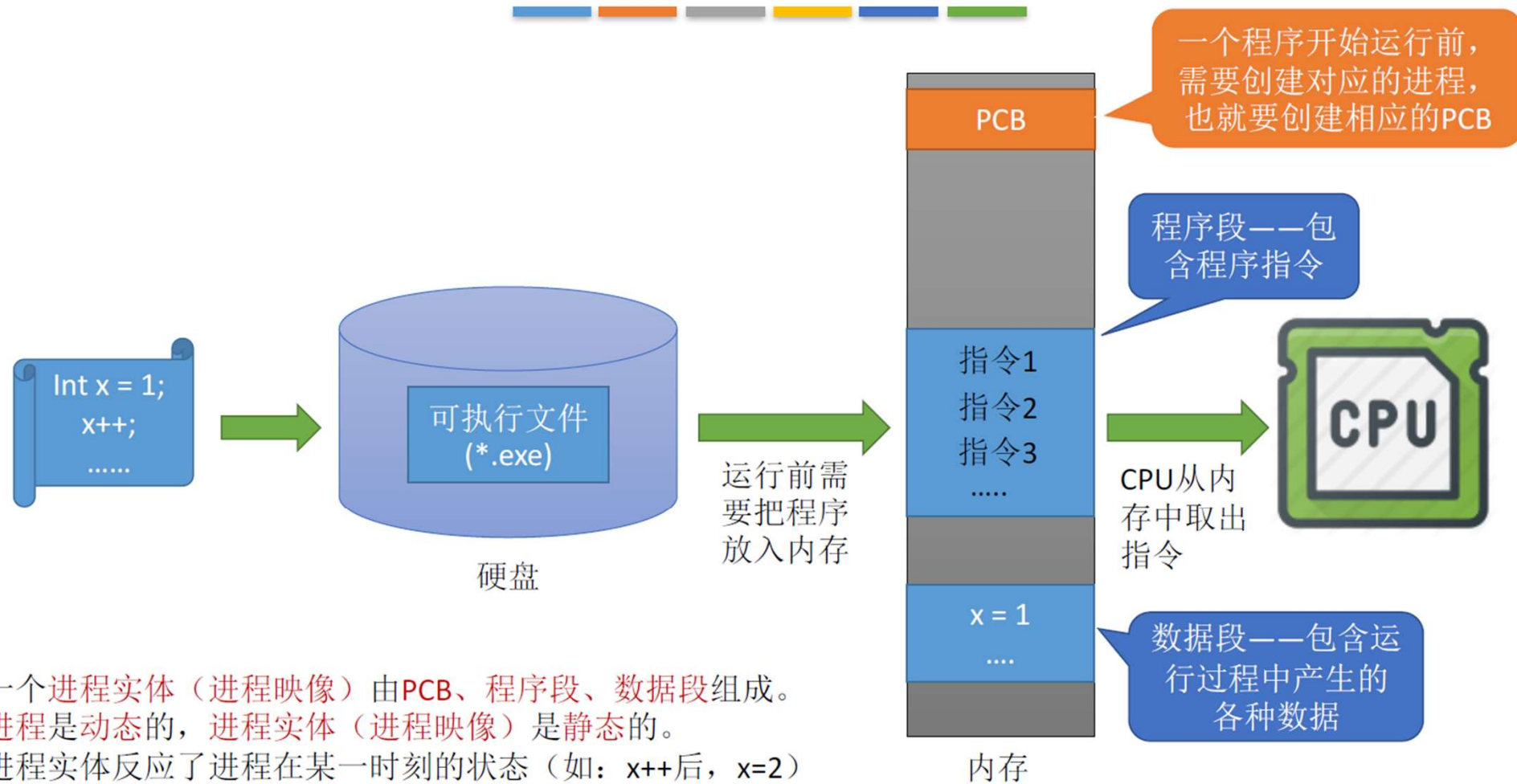
进程数据

: 运行过程中产生的各种数据

栈



## 知识滚雪球：程序是如何运行的？



一个进程实体（进程映像）由PCB、程序段、数据段组成。  
进程是动态的，进程实体（进程映像）是静态的。  
进程实体反应了进程在某一时刻的状态（如：`x++`后，`x=2`）

# 进程要素

## 1 构成

### 进程控制块PCB (进程属性)

- ❏ 处于核心段(内核)
- ❏ 用户进程不能直接访问、修改自身的PCB

# 进程要素

## 2 进程上下文

(1) 进程上下文：进程本身 + 运行环境  
对进程执行活动全过程的静态描述

由进程的用户地址空间内容、硬件寄存器内容及与该进程相关的核心数据结构组成

# 进程要素

## 2 进程上下文

❖ 寄存器级上下文：PSW寄存器、处理器状态寄存器、栈指针、通用寄存器的值

❖ 系统级上下文：

静态部分：PCB和资源表格

动态部分：核心栈（核心过程的栈结构，不同进程在调用相同核心过程时有不同核心栈）

❖ 用户级上下文：进程的用户地址空间，包括用户正文段、用户数据段和用户栈

进程的CPU现场信息

CPU中信息

内存中信息

# 进程要素

## 2 进程上下文

(2) 进程上下文切换 = 进程在系统中的调度、切换

- ❏ 保存被中断进程的处理器现场信息
- ❏ 修改被中断进程的进程控制块的有关信息，如进程状态等
- ❏ 把被中断进程的进程控制块加入有关队列
- ❏ 选择下一个占有处理器运行的进程

## 2 进程上下文

### (2) 进程上下文切换

- ❏ 修改被选中进程的进程控制块的有关信息
- ❏ 根据被选中进程设置操作系统用到的地址转换和存储保护信息
- ❏ 根据被选中进程恢复处理器现场

完成一个进程切换出CPU，  
另外一个进程切换进CPU  
的过程

# 进程及其实现

## 本讲内容

1. 进程定义
2. 进程的类型和特性
3. 进程的状态和转换
4. 进程控制块
5. 进程要素