

考试范围:

1. 进程管理

- 进程
- 线程
- CPU 调度
- 进程同步
- 死锁

3 道题

2. 内存管理

- 内存
- 虚拟内存

2 道题

3. 存储管理

- 文件系统
- 文件系统实现

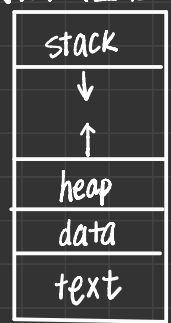
1 道题

4. 分布式系统

1 道题

# 进程:

内存中的进程



进程状态变化:



{  
 program → passive  
 process → active

进程控制块 (PCB) →

{  
 process state  
 PC  
 cpu register  
 cpu scheduling information  
 ...

↓  
 在linux中的task\_struct.  
 存储了该进程的所有信息

调度程序 → job scheduler → 从磁盘到内存  
 ↓  
 cpu scheduler → 从就绪队列到CPU.

o 父子进程具有相同的地址空间

{  
 zombie process ... 子进程终止但父进程尚未调用 wait()  
 orphan process ... 父进程没有调用 wait() 就终止, 父进程改为 init

{  
 shared-memory (生产者-消费者问题) ... 共用一个PIPE ↓  
 message-passing ( send(...), receive(...) )  
 写时先关闭读口, 写完关闭写口.  
 读时先关闭写口, ...

# deadlock

## ① 特点.

预防: (只要不满足这四个条件就不会)

- mutual exclusion — 不可避免
- Hold and wait — : 当一个进程申请资源时, 保证不持有其它
- No preemption — 可以抢占 waiting 状态的进程资源
- circular wait. — 为不同资源分配优先级 ↑

## ② 死锁避免.

在分配资源前考虑分配后是否处于 safe state.

- resource - allocation graph
- banker's algorithm