

# 操作系统概论第九节课笔记

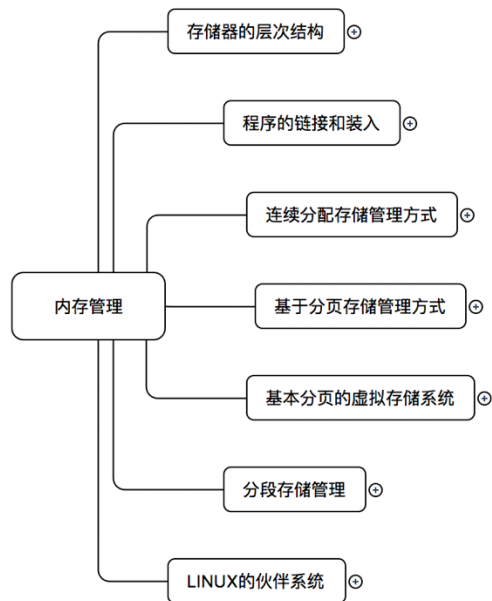
## 目录

- 一、 课件下载及重播方法
- 二、 本章知识点结构图
- 三、 本章知识点及考频总结
- 四、 配套练习题

### 一、 课件下载及重播方法



## 二、本章知识点结构图



### 三、本章知识点及考频总结

#### (一) 选择题 (共 12 道)

1. 缺页率对有效访问时间的影响 :  $P$  为缺页率 , 有效访问时间  $= 0.1 + 24999.9 \times P$  , 有效访问时间与缺页率成正比 , 缺页率越高 , 有效访问时间越长 , 访问效率越低。
2. 工作集 : 某段时间间隔里 , 进程实际要访问的页的集合。
3. 引入工作集目的 : 降低缺页率 , 提高访问内存效率。
4. 抖动 : 运行进程的大部分时间都用于页的换入换出 , 几乎不能完成任何有效果工作的状态。
5. 产生原因 : 进程数量太多 , 分配页框太少。
6. 预防方法 : ( 1 ) 采取局部置换策略 ; ( 2 ) 引入工作集 ; ( 3 ) 挂起若干进程。
7. 在分段存储管理的系统中 , 程序员使用二维的逻辑地址 , 一个数用来表示段 ,

另一个数用来表示段内偏移。

8. 引入分段机制的优点：方便编程、分段共享、分段保护、动态链接，以及动态空间的生长。

9. 分段的概念：进程的地址空间被划分成若干个段；每个段定义了一组逻辑信息，每个段的大小由相应的逻辑信息组的长度确定，段的大小不一样，每个段的逻辑地址从0开始，采用一段连续的地址空间；系统为每个段分配一个连续的物理内存区域，各个不同的段可以离散地放入物理内存不同的区域；系统为每个进程建立一张段表，段表的每一个表项纪录的信息包括段号、段长和该段的基址，段表存放在内存中。

10. 段表是由操作系统维护的用于支持分段存储管理地址映射的数据结构。

11. 每个进程有一个段表，段表由段表项构成。每个段表项包括段号、段基址（段的起始地址）和段长（段大小）3个部分。

12. 分段系统的地址变换：若已知逻辑单元的地址为s:d，求相应物理地址的步骤如下：1）以段号作索引，从段表中找到段号为s的段表项；

2）从找到的段表项中读出s段的基址和段大小；

3）如果 $d \leq \text{段大小}$ ，则将段基址与段内偏移d相加，得到与逻辑单元s:d相应的物理单元地址。

13. 分页和分段的主要区别：分页和分段都属于离散分配方式，都要通过数据结构与硬件的配合来实现逻辑地址到物理地址的映射。1）页是按物理单位划分的，

分页的引入是为了提高内存的利用率和支持虚拟存储 ;而段是按逻辑单位划分的 ,一个段含有一组意义相对完整的信息 ,引入分段的目的是为了更方便程序员编程。

2 ) 页的大小是固定的 ,而段的大小不固定 ,取决于用户编写的程序 ;3 ) 分页的地址空间是一维的 ,分段的地址空间是二维的。

14. 段页式存储管理的基本原理 :将用户进程的逻辑空间先划分成若干个段 ,每个段再划分成若干个页。进程以页为单位在物理内存中离散存放 ,每个段中被离散存放的页具有逻辑相关性。为了实现地址映射 ,操作系统为每个进程建立一个段表 ,再为每个段建立一个页表。进程段表的每一个段表项存放某个段的页表起始地址和页表长度。

15. 段页式地址变换过程 :1 ) 以段号s作索引 ,找到段s的段表项 ,得到该段页表的起始地址 ;2 ) 通过分页机制从段内偏移d中分离出页号P和页内偏移W ;3 ) 以段内页号P作索引 ,从段s的页表中搜索页号P对应的页表项 ;4 ) 从页表项得到页所在的页框号 ;5 ) 由页框号与页内偏移W得到对应的物理地址。物理地址=页框号\*页框大小+页内偏移。

16. 满足以下条件的两个块称为伙伴 :1 ) 两个块具有相同的大小 ,记作b ;2 ) 它们的物理地址是连续的 ,起始地址是2b的整数倍。

## (二) 填空题 (共13道)

1. 缺页率对有效访问时间的影响 :P为缺页率 ,有效访问时间= $0.1 + 24999.9 \times P$  ,有效访问时间与缺页率成正比 ,缺页率越高 ,有效访问时间越长 ,访问效率越低。

2. 工作集 :某段时间间隔里 ,进程实际要访问的页的集合。

3. 引入工作集目的：降低缺页率，提高访问内存效率。
4. 抖动：运行进程的大部分时间都用于页的换入换出，几乎不能完成任何有效果工作的状态。
5. 产生原因：进程数量太多，分配页框太少。
6. 预防方法：(1) 采取局部置换策略；(2) 引入工作集；(3) 挂起若干进程。
7. 在分段存储管理的系统中，程序员使用二维的逻辑地址，一个数用来表示段，另一个数用来表示段内偏移。
8. 引入分段机制的优点：方便编程、分段共享、分段保护、动态链接，以及动态空间的生长。
9. 分段的概念：进程的地址空间被划分成若干个段；每个段定义了一组逻辑信息，每个段的大小由相应的逻辑信息组的长度确定，段的大小不一样，每个段的逻辑地址从0开始，采用一段连续的地址空间；系统为每个段分配一个连续的物理内存区域，各个不同的段可以离散地放入物理内存不同的区域；系统为每个进程建立一张段表，段表的每一个表项纪录的信息包括段号、段长和该段的基址，段表存放在内存中。
10. 段表是由操作系统维护的用于支持分段存储管理地址映射的数据结构。
11. 每个进程有一个段表，段表由段表项构成。每个段表项包括段号、段基址(段的起始地址)和段长(段大小)3个部分。
12. 分段系统的地址变换：若已知逻辑单元的地址为s:d，求相应物理地址的步骤

如下：1) 以段号作索引，从段表中找到段号为s的段表项；

2) 从找到的段表项中读出s段的基地址和段大小；

3) 如果 $d \leq \text{段大小}$ ，则将段基址与段内偏移d相加，得到与逻辑单元s:d相应的物理单元地址。

13. 分页和分段的主要区别：分页和分段都属于离散分配方式，都要通过数据结构与硬件的配合来实现逻辑地址到物理地址的映射。1) 页是按物理单位划分的，分页的引入是为了提高内存的利用率和支持虚拟存储；而段是按逻辑单位划分的，一个段含有一组意义相对完整的信息，引入分段的目的是为了更方便程序员编程。

2) 页的大小是固定的，而段的大小不固定，取决于用户编写的程序；3) 分页的地址空间是一维的，分段的地址空间是二维的。

14. 段页式存储管理的基本原理：将用户进程的逻辑空间先划分成若干个段，每个段再划分成若干个页。进程以页为单位在物理内存中离散存放，每个段中被离散存放的页具有逻辑相关性。为了实现地址映射，操作系统为每个进程建立一个段表，再为每个段建立一个页表。进程段表的每一个段表项存放某个段的页表起始地址和页表长度。

15. 段页式地址变换过程：1) 以段号s作索引，找到段s的段表项，得到该段页表的起始地址；2) 通过分页机制从段内偏移d中分离出页号P和页内偏移W；3) 以段内页号P作索引，从段s的页表中搜索页号P对应的页表项；4) 从页表项得到页所在的页框号；5) 由页框号与页内偏移W得到对应的物理地址。物理地址=页框号\*页框大小+页内偏移。

16. 满足以下条件的两个块称为伙伴：1) 两个块具有相同的大小，记作 $b$ ；2) 它们的物理地址是连续的，起始地址是 $2b$ 的整数倍。

#### 四、配套练习题

1. 为每个页记录该页调入内存的时间，当选择换出页时，选择进入内存时间最早的页，这种算法是（ ）

- |              |                  |
|--------------|------------------|
| A. 最佳置换算法    | B. 先进先出算法        |
| C. 最近最久未使用算法 | D. 简单 Clock 置换算法 |

2. 选择在最近的过去最久未访问的页面予以置换的算法是（ ）

- |        |          |
|--------|----------|
| A. ORA | B. FIFO  |
| C. LRU | D. Clock |

3. 能引起抖动系统抖动的主要原因是（ ）

- |           |            |
|-----------|------------|
| A. 进程数量太少 | B. 进程数量不固定 |
| C. 进程数量固定 | D. 进程数量太多  |

【参考答案】BCD