1. 动态内存分配需要对内存分区进行管理，一般使用位图和空闲链表两种方法。128MB 的内存以n字节为单元分配，对于链表，假设内存中数据段和空闲区交替排列，长度均 为64KB。并假设链表中的每个节点需要记录32位的内存地址信息、16位长度信息和16 位下一节点域信息。这两种方法分别需要多少字节的存储空间？那种方法更好？

* 128 MB = 227字节。对于位图，用于存储管理需要227/8n字节，故总共需要 227+227/8n=227×(1+1/8n)字节；对于链表，用于存储管理需要 227/ 216(64KB)=211个节点，每个节点大小为需要(32+16+16)/8 = 8字节，故总共需要227+ 211×8= 227+ 214=227×(1 +1/(8×210) )字节；因此，当n <210字节（即1KB）时，位图>链表，则使用链表；当n > 1KB 时，位图 < 链表，则使用位图。

1. 在一个交换系统中，按内存地址排列的空闲区大小是: 10KB、4KB、20KB、18KB、7KB、 9KB、12KB和15KB。对于连续的段请求：12KB、10KB、9KB。使用FirstFit、BestFit、 WorstFit和NextFit将找出哪些空闲区？

* FirstFit ：20KB，10KB，18KB
* BestFit ：12KB，10KB，9KB
* WorstFit ：20KB，18KB，15KB
* NextFit ：20KB，18KB，9KB

1. 解释逻辑地址、物理地址、地址映射，并举例说明

* 逻辑地址：有地址变换功能的计算机中,访问指令给出的地址叫逻辑地址
* 物理地址：计算机物理内存中的实际地址称为物理地址
* 地址映射：将逻辑地址转换为物理地址的过程称之为地址映射
* 比如编程的时候，新建一个数组，数组第一个元素的逻辑地址是0，但是物理地址就是计算机为他分配的地址空间0x0000cdk3，当你使用数组的后几个元素的时候，物理地址相应的加起来就可以。

1. 解释页式（段式）存储管理中为什么要设置页（段）表和快表，简述页式（段式）地 址转换过程。

* 设快表是为了减少cpu访问内存的次数，从而减少程序运行的时间。系统先将页号与块表的表项进行比对，如果发现匹配，那么就直接从块表中取出块号。若是不匹配，则需要访问页表，同时依据不同的方式更新快表。

1. 叙述缺页中断的处理流程。

* 首先判断内存中有无空白页，若没有则依据某些方式淘汰一页，之后填写页表和存储分块表中的项，判断被淘汰的页是否被修改过。若是修改过，将该页写到外存。如果有空白页则选取一页空白页。之后根据外存所需的页号读取虚存，填写页表和存储分块表中的项。

1. 假设一个机器有38位的虚拟地址和32位的物理地址。

(1) 与一级页表相比，多级页表的主要优点是什么？

* 避免把全部页表保留在内存中。

(2) 如果使用二级页表，页面大小为16KB，每个页表项有4个字节。应该为虚拟地址中的第 一级和第二级页表域各分配多少位？

* 238/214 = 224，故页面长度为224，需要24位偏移量，而二级页表的表项为4字节，故PT2 = 2 ，所以 PT1 = 38 - 2 - 24 = 12，因此，对第一级页表域分配12位，对第二级页表分配域2位。

1. 假设页面的访问存在一定的周期性循环，但周期之间会随机出现一些页面的访问。例 如：0,1,2…,511,431,0,1,2…511,332,0,1,2,…,511等。请思考：

(1) LRU、FIFO和Clock算法的效果如何？

* 三种算法产生的缺页中断是一样的

(2) 如果有500个页框，能否设计一个优于LRU、FIFO和Clock的算法

* 如果分配了500个页框，那么0~498号页框是固定的，每次只有一个页框（第499页）进行页面置换。