

如果大部分进程在运行时都要增长，为了减少因内存区域不够而引起的进程交换和移动所产生的开销，一种可用的方法是，当换入或移动进程时为它分配一些额外的内存。然而，当进程被换出到磁盘上时，应该只交换进程实际上使用的内存中的内容，将额外的内存交换出去是一种浪费。在图3-5a中读者可以看到一种已为两个进程分配了增长空间的内存配置。

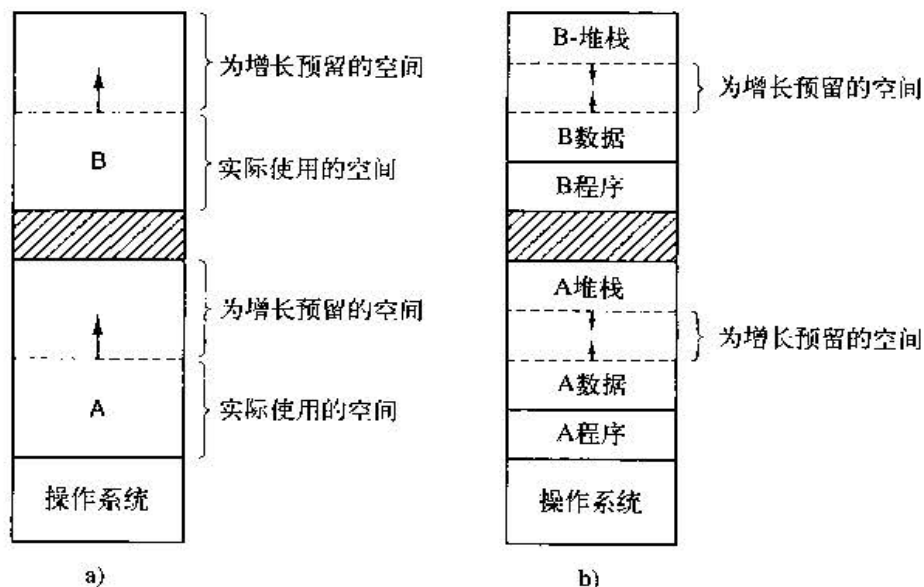


图3-5 a) 为可能增长的数据段预留空间；b) 为可能增长的数据段和堆栈段预留空间

如果进程有两个可增长的段，例如，供变量动态分配和释放的作为堆使用的一个数据段，以及存放普通局部变量与返回地址的一个堆栈段，则可使用另一种安排，如图3-5b所示。在图中可以看到所示进程的堆栈段在进程所占内存的顶端并向下降，紧接在程序段后面的数据段向上增长。在这两者之间的内存可以供两个段使用。如果用完了，进程或者必须移动到足够大的空闲区中（它可以被交换出内存直到内存中有足够的空间），或者结束该进程。

3.2.3 空闲内存管理

在动态分配内存时，操作系统必须对其进行管理。一般而言，有两种方式跟踪内存使用情况：位图和空闲链表。下面我们将介绍这两种方式。

1. 使用位图的存储管理

使用位图方法时，内存可能被划分成小到几个字或大到几千字节的分配单元。每个分配单元对应于位图中的一位，0表示空闲，1表示占用（或者相反）。一块内存区和其对应的位图如图3-6所示。

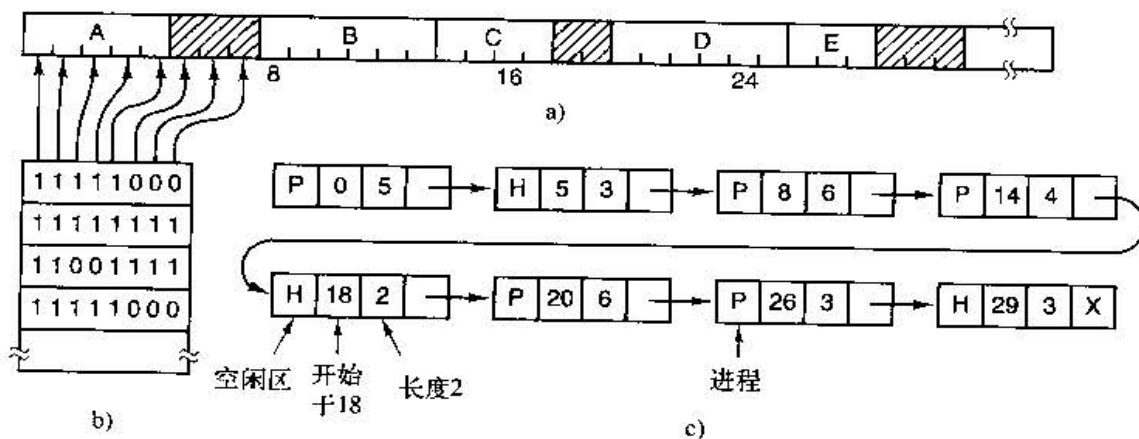


图3-6 a) 一段有5个进程和3个空闲区的内存，刻度表示内存分配的单元，阴影区域表示空闲（在位图中用0表示）；b) 对应的位图；c) 用空闲表表示的同样的信息