

作业 9

贾城昊

2021K8009929010

9.1 一台机器虚存采用分段机制，物理内存当前的空闲空间如下(按物理地址由小到大的顺序):12MB, 5MB, 18MB, 20MB, 8MB, 9MB, 10MB和15MB。此时要为三个段分配空间(按时间先后顺序): 段A申请12MB, 段B申请10MB, 段C申请9MB。请分别给出采用Best Fit, Worst Fit, First Fit 和 Next Fit算法下, 每次分配成的空闲空间状态(按物理地址由小到大顺序), 以及每次分配所需的比较次数。

解:

- Best Fit:

段A申请12MB: 比较1次

12MB

段B申请10MB: 比较6次

5MB, 18MB, 20MB, 8MB, 9MB, 10MB

段C申请9MB: 比较5次

5MB, 18MB, 20MB, 8MB, 9MB

- Worse Fit:

段A申请12MB: 比较8次

12MB, 5MB, 18MB, 20MB, 8MB, 9MB, 10MB,15MB

段B申请10MB: 比较8次

12MB, 5MB, 8MB, 8MB, 9MB, 10MB,15MB

段C申请9MB: 比较8次

12MB, 5MB, 8MB, 8MB, 8MB, 9MB, 10MB,15MB

- First Fit:

段A申请12MB: 比较1次

12MB

段B申请10MB: 比较2次

5MB, 18MB

段C申请9MB: 比较3次

5MB, 8MB, 20MB

- Next Fit:

段A申请12MB: 比较1次

12MB

段B申请10MB: 比较2次

5MB, 18MB

段C申请12MB: 比较1次

20MB

9.2 假设一台计算机使用 32-bit 的虚拟地址空间和三级页表，虚地址的划分为 8-bit | 6-bit | 6-bit | 12-bit（注：8 bit 对应为第一级页表的地址，以此类推），请计算：

（1）该计算机系统的页大小是多少？

（2）该三级页表一共能索引多少个页？

（3）现有一个程序的代码段大小为 8KB，数据段为 32KB，栈大小为 8KB，则在使用上述三级页表时，最少需要占用多少个物理页框？最多会占用多少个物理页框？（注：假设程序各段在地址空间中的布局可以自行决定）

（4）在上述（3）中，假设该计算机使用一级页表进行地址空间管理，则（3）中的程序需要占用多少个物理页框？

注：请写出计算过程。

解：

(1)

12-bit 用于页内偏移，所以页大小为 $2^{12} = 4096$ 字节，即 4KB。

(2)

第一级页表索引需要 8 位，第二级和第三级各需要 6 位。

因此，三级页表一共能索引的页数为 $2^8 \times 2^6 \times 2^6 = 2^{20}$ 个页。

(3)

代码段大小为 8KB，数据段大小为 32KB，栈大小为 8KB。

当所有的段地址连续时，且 4KB 对齐的时候，需要的物理页框数量最少。此时有 $\frac{8}{4} + \frac{32}{4} + \frac{8}{4} = 12$ 个页，所以只需要 1 个第三级页表，有 1 个第二级页表，1 个第一级页表，所以一共需要 $3 + 12 = 15$ 个物理页框

当程序各个段不按页对齐且每个段不连续时，需要的物理页框数量最多。此时代码段有 $\frac{8}{4} = 2$ 个页，不妨设其横跨 2 个第三级页表，且横跨 2 个第二级页表；数据段有 $\frac{32}{4} = 8$ 个页，不妨设其横跨 2 个第三级页表，且横跨 2 个第二级页表；栈有 $\frac{8}{4} = 2$ 个页，不妨设其横跨 2 个第三级页表，且横跨 2 个第二级页表。而第一级页表每个进程只对应一个，所以一共需要 $12 + 6 + 6 + 1 = 25$ 个物理页框。

综上，最少 15 个物理页框，最多 25 个物理页框

(注：上面最多情况的计算是认为每个段虚拟地址必须得连续，否则分成这些段没有意义；并且代码段，数据段，栈必须 4KB 对齐，因为页对齐通常是一个重要的性能和操作系统规范)

(4)

若该计算机使用一级页表进行地址空间管理，则一级页表有 4MB 的内容，需要 1K 个页，所以一共需要 $1024 + 12 = 1036$ 个物理页框

9.3 假设一台计算机上运行一个进程 A，该进程的地址空间大小为 4 MB（页大小为 4KB）。该计算机使用线性页表记录进程 A 的虚实映射关系，并且将 A 的页表都

保存在内存中。该计算机 CPU 的 TLB 大小为 32 项，每项 4B，一次 TLB 查询或 TLB 填充的延迟均为 5 ns，请计算：

(1) 假设该计算机使用软件处理 TLB miss，且操作系统进行一次页表查询的平均延迟为 100 ns，如果能让虚实地址映射的平均延迟为 40 ns，那么 TLB 的命中率应为多少？如果能让虚实地址映射的平均延迟不超过 20 ns，那么 TLB 的命中率应为多少？（上述各项操作的延迟不变）

解：

(1)

设 TLB 命中率为 x ，则平均延迟为

$$5x + (5 + 100 + 5 + 5) * (1 - x) = 115 - 110x \text{ ns}$$

所以要想平均延迟为 40ns，则解出 $x = 68.18\%$ ，即命中率为 68.18%

所以要想平均延迟为 20ns，则解出 $x = 86.36\%$ ，即命中率为 86.36%

9.4 现有如下 C 程序

```
uint32 X[N];
```

```
int step = M, i = 0;
```

```
for(i=0;i<N;i+=step) X[i] = X[i] + 1;
```

请计算：

(1) 假设该程序运行在一台计算机上，该计算机的虚址空间为 32-bit，物理地址空间为 2 GB，页大小为 4 KB，如果采用一级页表，则该页表的页表项一共有多少？

(2) 假设该计算机的 CPU 的 TLB 大小为 32 项，每项 4B，那么题述程序中的 M 和 N 取值为多少时，会使得程序中循环的每一次执行都会触发 TLB miss？（假设 TLB 初始为空）

(3) 在 (2) 中，M 和 N 取值多少时，会使得程序中的循环执行时 TLB hit 最多？（假设 TLB 初始为空）

解：

(1)

虚拟空间有 4GB，一个页为 4KB，所以页表项一共有 $\frac{4GB}{4KB} = 2^{20}$ 个页表项

(2)

1 个页大小为 4KB，要每次循环都触发 TLB miss，则要求 $4B \times M \geq 4KB$ ，所以 $M \geq 1024$ ，同时还要求 N 大于 TLB 中的表项，即 $N \geq 32$ 。

(3)

由于开始的时候，TLB 为空，所以至少会有一次 TLB miss，而要想只有一次 TLB miss，则 X 到 $X + (N - 1) \times M$ 都在一个页框里面，最理想的情况是 X 是 4KB 对齐的，那么只需要要求 $4 \times (N - 1) \times M \leq 4K$ ，此时可以取 $N = 1024$ ， $M = 1$ ，那么会 miss 一次而命中 1023 次。

所以只需要 N 为 1024 的倍数且 $M=1$ 的时候，每次 1024 次循环都只会有一次 TLB miss 而命中 1023 次，

特别的，如果 $M = 1$ ， $N = 1024 \times 32$ ，则会 miss 32 次而命中 1023×32 次，此时 TLB 已经完全装载好，若之后 TLB 表项没有被替换，继续运行该循环，则永远不会 TLB miss 了