9.11：

地址翻译的详细文档[52]。

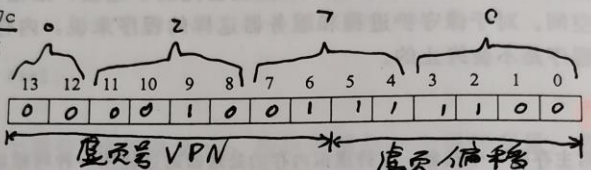
Knuth 在 1968 年编写了有关内存分配的经典之作[64]。从那以后，在这个领域就有了大量的文献。Wilson、Johnstone、Neely 和 Boles 编写了一篇关于显式分配器的漂亮综述和性能评价的文章[118]。本书中关于各种分配器策略的吞吐率和利用率的一般评价就引自于他们的调查。Jones 和 Lins 提供了关于垃圾收集的全面综述[56]。Kernighan 和 Ritchie [61]展示了一个简单分配器的完整代码，这个简单的分配器是基于显式空闲链表的，每个空闲块中都有一个块大小和后继指针。这段代码使用联合(union)来消除大量的复杂指针运算，这是很有趣的，但是代价是释放操作是线性时间(而不是常数时间)。Doug Lea 开发了广泛使用的开源 malloc 包，称为 dlmalloc [67]。

家庭作业

•9.11 在下面的一系列问题中，你要展示 9.6.4 节中的示例内存系统如何将虚拟地址翻译成物理地址，以及如何访问缓存。对于给定的虚拟地址，请指出访问的 TLB 条目、物理地址，以及返回的缓存字节值。请指明是否 TLB 不命中，是否发生了缺页，是否发生了缓存不命中。如果有缓存不命中，对于“返回的缓存字节”用“-”来表示。如果有缺页，对于“PPN”用“-”来表示，而 C 部分和 部分就空着。

虚拟地址：0x027c

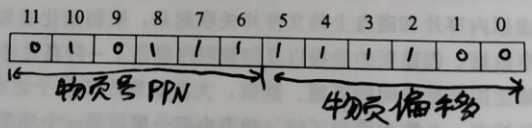
A. 虚拟地址格式



B. 地址翻译

参数	值
VPN	0x09
TLB索引	0x01
TLB标记	0x02
TLB命中？（是/否）	No
缺页？（是/否）	No
PPN	0x17

C. 物理地址格式



D. 物理地址引用

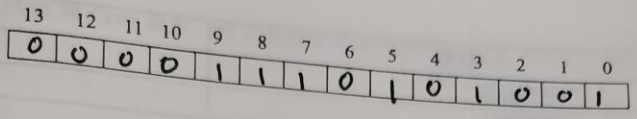
参数	值
字节偏移	0x00
缓存索引	0x0F
缓存标记	0x17
缓存命中？（是/否）	No
返回的缓存字节	无

•9.12 对于下面的地址，重复习题 9.11：

虚拟地址：0x03a9

A. 虚拟地址格式

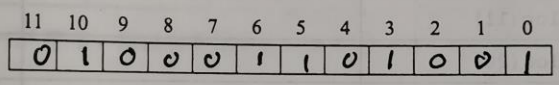
9.12：



B. 地址翻译

参数	值
VPN	0x0E
TLB索引	0x02
TLB标记	0x03
TLB命中？（是/否）	No
缺页？（是/否）	No
PPN	0x11

C. 物理地址格式



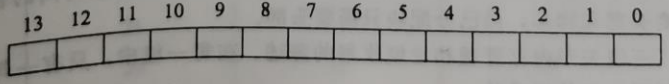
D. 物理地址引用

参数	值
字节偏移	0x01
缓存索引	0x0A
缓存标记	0x11
缓存命中？（是/否）	No
返回的缓存字节	无

9.13 对于下面的地址，重复习题9.11：

虚拟地址： 0x0040

A. 虚拟地址格式

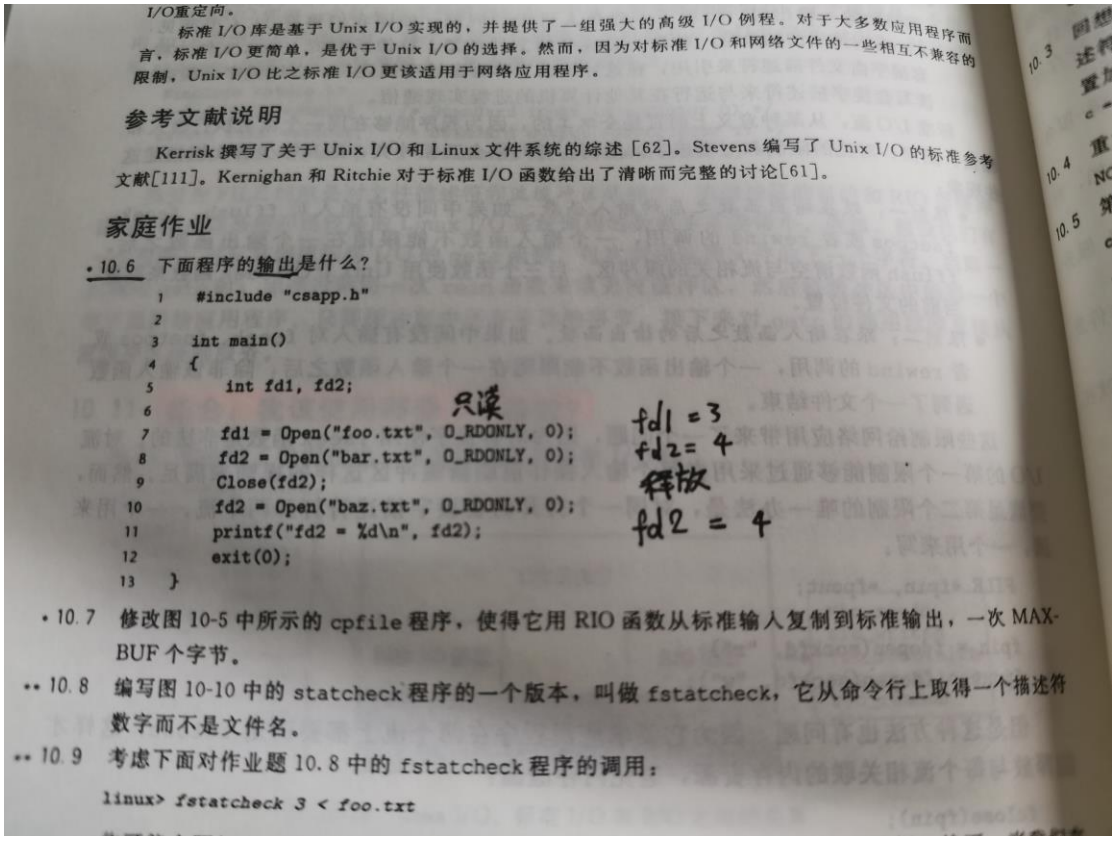


B. 地址翻译

参数	值
VPN	
TLB索引	
TLB标记	

10.6：

fd2 = 4



I/O重定向。

标准 I/O 库是基于 Unix I/O 实现的，并提供了一组强大的高级 I/O 例程。对于大多数应用程序而言，标准 I/O 更简单，是优于 Unix I/O 的选择。然而，因为对标准 I/O 和网络文件的一些相互不兼容的限制，Unix I/O 比之标准 I/O 更该适用于网络应用程序。

参考文献说明

Kerrisk 撰写了关于 Unix I/O 和 Linux 文件系统的综述 [62]。Stevens 编写了 Unix I/O 的标准参考文献[111]。Kernighan 和 Ritchie 对于标准 I/O 函数给出了清晰而完整的讨论[61]。

家庭作业

• 10.6 下面程序的输出是什么？

```
1   #include "csapp.h"
2
3   int main()
4   {
5       int fd1, fd2;
6
7       fd1 = Open("foo.txt", O_RDONLY, 0);
8       fd2 = Open("bar.txt", O_RDONLY, 0);
9       Close(fd2);
10      fd2 = Open("baz.txt", O_RDONLY, 0);
11      printf("fd2 = %d\n", fd2);
12      exit(0);
13  }
```

只读

fd1 = 3
fd2 = 4
释放
fd2 = 4

• 10.7 修改图 10-5 中所示的 cpfile 程序，使得它用 RIO 函数从标准输入复制到标准输出，一次 MAXBUF 个字节。

•• 10.8 编写图 10-10 中的 statcheck 程序的一个版本，叫做 fstatcheck，它从命令行上取得一个描述符数字而不是文件名。

•• 10.9 考虑下面对作业题 10.8 中的 fstatcheck 程序的调用：

```
linux> fstatcheck 3 < foo.txt
```