Chapter 3

4. First fit takes 20 MB, 10 MB, and 18 MB.
Best fit takes 12 MB, 10 MB, and 9 MB.
Worst fit takes 20 MB, 18 MB, and 15 MB.
Next fit takes 20 MB, 18 MB, and 9 MB.

6. For a 4-KB page size the (page, offset) pairs are (4, 3616), (8, 0), and (14, 2656). For an 8-KB page size they are (2, 3616), (4, 0), and (7, 2656).

7. (a) 8212. (b) 4100. (c) 24684.

17. Consider,
(a) A multilevel page table reduces the number of actual pages of the page table that need to be in memory because of its hierarchic structure. In fact, in a program with lots of instruction and data locality, we only need the top-level page table (one page), one instruction page and one data page.
(b) Allocate 12 bits for each of the three page fields. The offset field requires 14 bits to address 16 KB. That leaves 24 bits for the page fields. Since each entry is 4 bytes, one page can hold 2^{12} page table entries and therefore requires 12 bits to index one page. So allocating 12 bits for each of the page fields will address all 2^{38} bytes.

19. Twenty bits are used for the virtual page numbers, leaving 12 over for the offset. This yields a 4-KB page. Twenty bits for the virtual page implies 2^{20} pages.

25. The main memory has $2^{18}/2^{13} = 32$ pages. A 32 hash table will have a mean chain length of 1. To get under 1, we have to go to the next size, 64 entries. Spreading 32 entries over 64 table slots will give a mean chain length of 0.5, which ensures fast lookup.

28. The page frames for FIFO are as follows:

Page	4 page frames				
Refs	Fault?	Page Contents			
0	yes	0			
1	yes	1	0		
7	yes	7	1	0	
2	yes	2	7	1	0
3	yes	3	2	7	1
2	no	3	2	7	1
7	no	3	2	7	1
1	no	3	2	7	1
0	yes	0	3	2	7
3	no	0	3	2	7

The page frames for LRU are as follows:

Page	4 page frames				
Refs	Fault?	Page Contents			
0	yes	0			

1	yes	1	0		
7	yes	7	1	0	
2	yes	2	7	1	0
3	yes	3	2	7	1
2	no	2	3	7	1
7	no	7	2	3	1
1	no	1	7	2	3
0	yes	0	1	7	2
3	yes	3	0	1	7

FIFO yields 6 page faults; LRU yields 7.

30. The counters are
Page 0: 01101110
Page 1: 01001001
Page 2: 00110111
Page 3: 10001011

33. Consider,
(a) For every R bit that is set, set the time-stamp value to 10 and clear all R bits. You could also change the (0,1) R-M entries to (0,0*). So the entries for pages 1 and 2 will change to:

Page	Time stamp	V	R	M
0	6	1	0	0*
1	10	1	0	0
2	10	1	0	1

(b) Evict page 3 (R = 0 and M = 0) and load page 4:

Page	Time stamp	V	R	M	Notes
0	6	1	0	1	
1	9	1	1	0	
2	9	1	1	1	
3	7	0	0	0	Changed from 7 (1,0,0)
4	10	1	1	0	Changed from 4 (0,0,0)

36. NRU removes page 2. FIFO removes page 3. LRU removes page 1. Second chance removes page 2.

38. Fragment *B* since the code has more spatial locality than Fragment *A*. The inner loop causes only one page fault for every other iteration of the outer loop. (There will only be 32 page faults.) Aside (Fragment *A*): Since a frame is 128 words, one row of the *X* array occupies half of a page (i.e., 64 words). The entire array fits into 64*32/128 = 16 frames. The inner loop of the code steps through consecutive rows of *X* for a given column. Thus every other reference to *X* [*i*][*j*] will cause a page fault. The total number of page faults will be 64 *64/2 = 2,048.