1. (20%; 10% each) Consider the two-dimensional array A:

double A[][] = new double[200][200];

where each double occupies 8 bytes and A[0][0] is at location 200, in a paged system with pages of size 200 bytes. A small process is in page 0 (locations 0 to 199) for manipulating the matrix; thus, every instruction fetch will be from page 0. For three page frames, how many page faults are generated by the following array initialization loops, using LRU replacement and assuming (1) page frame 0 has the process in it, (2) the other two are initially empty, and (3) the array is stored in memory column-major?

1. for (int i = 0; i < 200; i++) for (int j = 0; j < 200; j++) A[i][j] = 0;

Ans:

Each time(loop) loads a page in colum, but modifies in row.

200 \* 200 \*0.5= 20000

1. for (int j = 0; j < 200; j++) for (int i = 0; i < 200; i++) A[i][j] = 0;

Ans:

A page has 200 bytes, a double has 8 bytes. So, a page can contain 25 elements of array.

200/8=25

A column needs 8 page to contain.

200/25=8

Have 200 columns, each column needs 8 pages. Total needs 1600 pages.

200\*8=1600

1. (20%; 10% each) A disk has 16000 cylinders, each with 10 tracks of 512 blocks. A seek takes 1 ms per cylinder moved. If no attempt is made to put the blocks of a file close to each other, two blocks that are logically consecutive (i.e., follow one another in the file) will require an average seek, which takes 6 ms. If, however, the operating system makes an attempt to cluster related blocks, the mean interblock distance can be reduced to 2 cylinders and the seek time reduced to 200 µs. Assuming that the rotational latency is 3 ms and the transfer time is 10 µs *per block, answer the following questions.*

*(a) How long does it take to read a 200 block randomly placed file?*

*(b) How long does it take to read a 200 block clustered file*

*Ans*

*Need seek and rotate and transfer.*

*Random*

*(6+3+0.01)\*200=1802ms*

*Clustered*

*(0.2+3+0.01)\*200=682ms*

3.(10%) A computer has 6 GB of RAM allocated in units of 4 KB. How many KB are needed if a bit map is used to keep track of free memory?

Ans:

We need

6GB/4KB = 6\*2^20 KB / 4KB = 6\*2^18 bits = 6\*2^15bytes= 6\*2^5KB

4. (10%; 5% each) Assume a page reference string for a process with m frames (initially all empty). The page reference string has length n with p distinct page numbers occurring in it. For any page-replacement algorithms,

(a) What is a lower bound on the number of page faults?

Ans:

(load all pages) p

(b)What is an upper bound on the number of page faults?

Ans:

(each byte need a page to load in) n

5.(10%) Given a UNIX i-node with ten direct blocks and three levels of indirect blocks (i.e., a single, a double, and a triple) and assuming that the sizes of a pointer and a block are, respectively, 4 bytes and 4 Kbytes, what would be the size of the largest file allowed in bytes?

(a) (10%) What would be the size of the smallest file allowed in bytes?

0, because it can point to nothing.(which)

1個block => 8Kb

(b) (10%) What would be the size of the largest file allowed in bytes?

Ans:A pointer can point to 4K, the efficient will extend to 1024 times.

4Kb / 4 = 1K

1K \* 1K = 1M

1M \* 1M = 1G

=> ( 10 + 1K + 1M + 1G) \* 4Kb

6. . (10%) A computer whose processes have 1024 pages in their address spaces keeps its page tables in

memory. The overhead required for reading a word from the page table is 600 nsec. To reduce this

overhead, the computer has a TLB, which holds 32 (page, frame) pairs and can do a lookup in 100 nsec.

What hit rate is needed to reduce the mean overhead to 200 nsec or less?

Ans:

Treat it as expection, suppose hit rate(probability) is p

100p+(500+100)(1-p)<=200

* P>=0.8

7. 7. (20%; 5% each) Suppose that a disk drive has 1000 cylinders, numbered from 0 to 999. The drive is currently serving a request at cylinder 200, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is

50, 500, 250, 800, 350, 550, 400, 600, 100.

Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests for each of the following disk scheduling algorithms?

1. SCAN

SCAN(go to last block of disk, then service back to begin block of disk)

(125)、200、250、350、400、500、550、600、800、999、100、50

total dist = 50 + 100 + 50 + 100 + 50 + 50 + 200 + 199 + 899 + 50 = 1748

1. LOOK

LOOK(go to biggest/samllest request, then service back)

(125)、200、250、350、400、500、550、600、800、100、50

total dist = 50 + 100 + 50 + 100 + 50 + 50 + 200 + 700 + 50 = 1350

1. C-SCAN

C-SCAN(not service back)

(125)、200、250、350、400、500、550、600、800、999、0、50、100

total dist = 50 + 100 + 50 + 100 + 50 + 50 + 200 + 199 + 999 + 50 + 50 = 1898

1. C-LOOK

C-LOOK(not servece back)

(125)、200、250、350、400、500、550、600、800、50、100

total dist = 50 + 100 + 50 + 100 + 50 + 50 + 200 + 750 + 50 = 1400

(e) SSTF(shortest seek first)

200 -> 250 -> 350 -> 400 -> 500 -> 550 -> 600 -> 800 -> 100 -> 50

Total distance: 50 + 100 + 50 + 100 + 50 + 50 + 200 + 700 + 50 = 1350

**8.**A small computer has 8 page frames, each containing a page. The page frames contain virtual pages A, C, G, H, B, L, N, and D in that order. Their respective load times were 18, 23, 5, 7, 32, 19, 3, and 8. Their reference bits are 1, 0, 1, 1, 0, 1, 1, and 0 and their modified bits are 1, 1, 1, 0, 0, 0, 1, and 1, respectively. Which page will the second chance page replacement algorithm replace?

D.

Construct the FIFO queue first.

N -> G-> H -> D -> A -> L -> C -> B

Consider the algorithm:

Pages in list are sorted in FIFO order

R bits are cleared regularly

If the R bit of the oldest page is set it is put at the end of the list

If all the pages in the list have been referenced the page that was “recycled” will reappear with the R bit cleared and will be thrown away.

Page 0, because the page 0 R bit is 0, and the second-earliest page loaded. The page 2’s R bit is 1, will put at the end of the list by FIFO replace scheduler.

Then we can find D.

=> N(1,1) => N(0,1)

=> G(1,1) => G(0,1)

=> H(1,0) => H(0,0)

=> D(0,1) choose

**8.1 .** (10%) A small computer has 8 page frames, each containing a page. The page frames contain virturl pages A, C, G, H, B, L, N, and D in that order. Their respective load times were 18, 23, 5, 7, 32, 19, 3,and 8. Their reference bits are 1, 0, 1, 1, 0, 1, 1, and 1 and their modified bits are 1, 1, 1, 0, 0, 0, 1, and 1,respectively. Which page will the second chance page replacement algorithm replace?

=> N(1,1) => N(0,1)

=> G(1,1) => G(0,1)

=> H(1,0) => H(0,0)

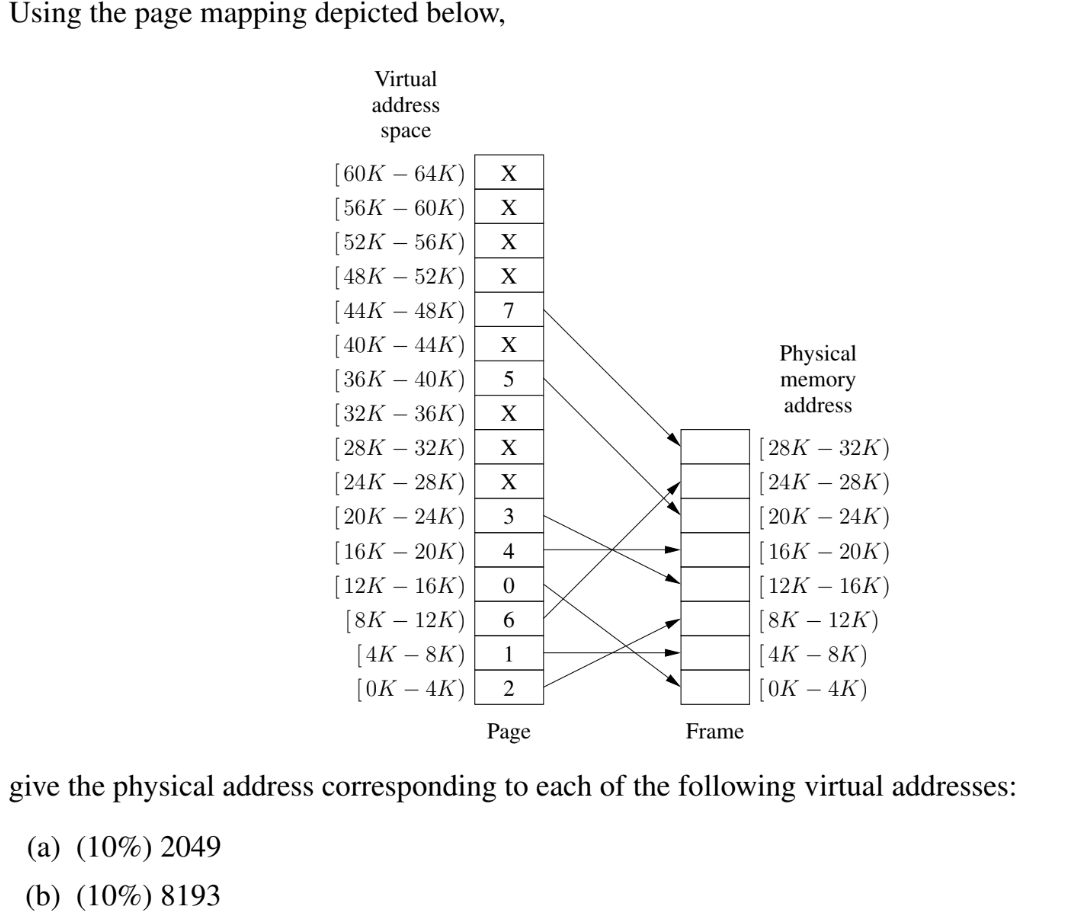
=> D(1,1) => D(0,1)

=> A(1,1) => A(0,1)

=> L(1,0) => L(0,0)

=> C(0,1) choose

**9**



1. virtual address 2049 in page 0 ->frame 2(8k-12k),so offset=2049-(0\*4k)=2049 8k+2049=8192+2049=10241
2. virtual address 8193 in page 2 ->frame 6(24k-28k),so offset = 8193-(2\*4K) = 1

24k+1=24577

10.Disk requests come in to the driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 5 msec per cylinder moved. How much seek time is needed for

(a) (10%) Closest cylinder next, and

(b) (10%) Elevator algorithm (initially moving upward).

In all cases, the arm is initially at cylinder 20.

(a)

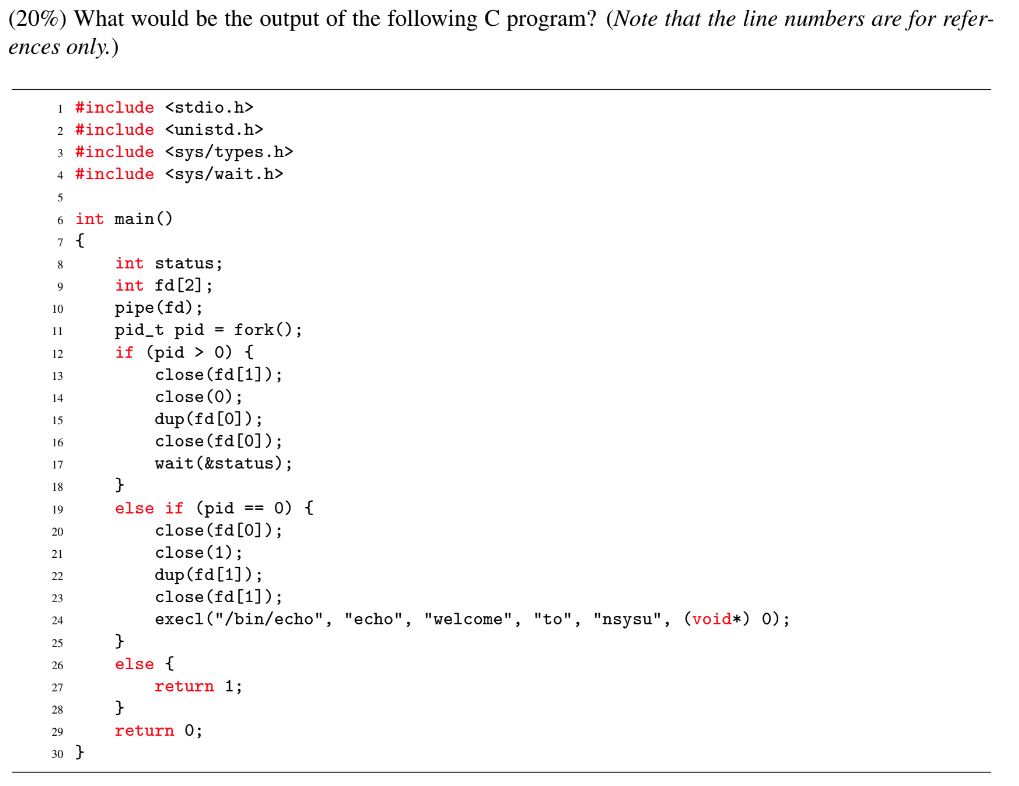
(20)、22、10、6、2、38、40

=> (2 + 12 + 4 + 4 + 36 + 2)\*5 = 300ms

(b)

(20)、22、38、40、10、6、2

=> (2 + 16 + 2 + 30 + 4 + 4)\*5 = 290ms

11. 

No output.

Because the parent process get the IPC in STDIN buffer, but not recieve it in or process something out.

11.(10%) A computer has six tape drives, with n processes competing for them. Each process may need two drives. For which values of n is the system deadlock free

Ans:

N<6

With three proccesses, each one can have two drives. With four processes, the distribution of drives will be (2,2,1,1), allowing the first two processes to finsh. With five processes, the distribution of drives will be (2,1,1,1,1). which still allows the first one to finish. With six processes, each holding one tape drive, and wanting another one, we have a deadlock. Thus for n < 6, the system is deadlock free.

12. 6. Consider the following page reference string:

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6.

How many page faults would occur for the following replacement algorithms, assuming three, four, or five frames? Remember all frames are initially empty, so your first unique pages will all cost one fault each.

(a) (10%) LRU replacement

(b) (10%) Optimal replacement

LRU replacement:

An approximation of optimal algorithm,looking backward, rather than forward

It replaces the page that has not been used for the longest period of time

Optimal replacement:

Replace the page that will not be used for the longest period of time

LRU-3frames (15’s fault)

[ 1 ]F

[ 2 1 ]F

[ 3 2 1 ]F

[ 4 3 2 ]F

[ 2 4 3 ]

[ 1 2 4 ]F

[ 5 1 2 ]F

[ 6 5 1 ]F

[ 2 6 5 ]F

[ 1 2 6 ]F

[ 2 1 6 ]

[ 3 2 1 ]F

[ 7 3 2 ]F

[ 6 7 3 ]F

[ 3 6 7 ]

[ 2 3 6 ]F

[ 1 2 3 ]F

[ 2 1 3 ]

[ 3 2 1 ]

[ 6 3 2 ]F

LRU-4frames (10’s fault)

[ 1 ]F

[ 2 1 ]F

[ 3 2 1 ]F

[ 4 3 2 1 ]F

[ 2 4 3 1 ]

[ 1 2 4 3 ]

[ 5 1 2 4 ]F

[ 6 5 1 2 ]F

[ 2 6 5 1 ]

[ 1 2 6 5 ]

[ 2 1 6 5 ]

[ 3 2 1 6 ]F

[ 7 3 2 1 ]F

[ 6 7 3 2 ]F

[ 3 6 7 2 ]

[ 2 3 6 7 ]

[ 1 2 3 6 ]F

[ 2 1 3 6 ]

[ 3 2 1 6 ]

[ 6 3 2 1 ]

LRU-5frames (8’s fault)

[ 1 ]F

[ 2 1 ]F

[ 3 2 1 ]F

[ 4 3 2 1 ]F

[ 2 4 3 1 ]

[ 1 2 4 3 ]

[ 5 1 2 4 3 ]F

[ 6 5 1 2 4 ]F

[ 2 6 5 1 4 ]

[ 1 2 6 5 4 ]

[ 2 1 6 5 4 ]

[ 3 2 1 6 5 ]F

[ 7 3 2 1 6 ]F

[ 6 7 3 2 1 ]

[ 3 6 7 2 1 ]

[ 2 3 6 7 1 ]

[ 1 2 3 6 7 ]

[ 2 1 3 6 7 ]

[ 3 2 1 6 7 ]

[ 6 3 2 1 7 ]

optimal-3frames (11’s fault)

[ 1 ]F

[ 1 2 ]F

[ 1 2 3 ]F

[ 1 2 4 ]F

[ 1 2 4 ]

[ 1 2 4 ]

[ 1 2 5 ]F

[ 1 2 6 ]F

[ 1 2 6 ]

[ 1 2 6 ]

[ 1 2 6 ]

[ 3 2 6 ]F

[ 3 7 6 ]F

[ 3 7 6 ]

[ 3 7 6 ]

[ 3 2 6 ]F

[ 3 2 1 ]F

[ 3 2 1 ]

[ 3 2 1 ]

[ 3 2 6 ]F

optimal-4frames (8’s fault)

[ 1 ]F

[ 1 2 ]F

[ 1 2 3 ]F

[ 1 2 3 4 ]F

[ 1 2 3 4 ]

[ 1 2 3 4 ]

[ 1 2 3 5 ]F

[ 1 2 3 6 ]F

[ 1 2 3 6 ]

[ 1 2 3 6 ]

[ 1 2 3 6 ]

[ 1 2 3 6 ]

[ 7 2 3 6 ]F

[ 7 2 3 6 ]

[ 7 2 3 6 ]

[ 7 2 3 6 ]

[ 1 2 3 6 ]F

[ 1 2 3 6 ]

[ 1 2 3 6 ]

[ 1 2 3 6 ]

optimal-5frames (7’s fault)

[ 1 ]F

[ 1 2 ]F

[ 1 2 3 ]F

[ 1 2 3 4 ]F

[ 1 2 3 4 ]

[ 1 2 3 4 ]

[ 1 2 3 4 5 ]F

[ 1 2 3 4 6 ]F

[ 1 2 3 4 6 ]

[ 1 2 3 4 6 ]

[ 1 2 3 4 6 ]

[ 1 2 3 4 6 ]

[ 1 2 3 7 6 ]F

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

[ 1 2 3 7 6 ]

13. (2011.5)A small computer has four page frames. At the first clock tick, the R bits are 0111 (page 0 is 0, the rest are 1). At subsequent clock ticks, the values are 1011, 1010, 1101, 0010, 1010, 1100, 0001, 0101, 1011, and 1101. If the aging algorithm is used with a 5-bit counter, give the values of the four counters after the last ticks.

* page 0: 11001
* page 1: 10101
* page 2: 01000
* page 3: 11110

14.(20%) A computer has four page frames. The time of loading, time of last access, and the R and M bits for each page are as shown below (the times are in clock ticks):

| **Page** | **Loaded** | **Last Ref.** | **R** | **M** |
| --- | --- | --- | --- | --- |
| 0 | 126 | 279 | 0 | 0 |
| 1 | 230 | 260 | 1 | 0 |
| 2 | 120 | 272 | 1 | 1 |
| 3 | 160 | 280 | 1 | 1 |

1. Which page will FIFO replace?

Page 2

Because the page 2 is the oldest loaded page.(in the FIFO queue)

1. Which page will LRU replace?

Page 1

Because the page 1’s last reference time is 260, which is the oldest referenced time.

Explain:

The least recently used (LRU) page replacement algorithm, though similar in name to NRU, differs in the fact that LRU keeps track of page usage over a short period of time, while NRU just looks at the usage in the last clock interval. LRU works on the idea that pages that have been most heavily used in the past few instructions are most likely to be used heavily in the next few instructions too.

1. Which page will NRU replace?

Page 0

Because it’s RM = 0.

(d) Which page will second chance replace?

Page 0

Because the page 0 R bit is 0, and the second-earliest page loaded. The page 2’s R bit is 1, will put at the end of the list by FIFO replace scheduler.

Explain page 19:

Pages in list are sorted in FIFO order

R bits are cleared regularly

If the R bit of the oldest page is set it is put at the end of the list

If all the pages in the list have been referenced the page that was “recycled” will reappear with the R bit cleared and will be thrown away.