CS & IT ENGINEERING

Operating Systems

1500 Series

Lecture No.- 04

By- Dr. Khaleel Khan sir

Recap of Previous Lecture

Topic Questions Practice

Topics to be Covered

Topic

Topic

Topic

Topic

Topic

Dining-Philosophers

[NAT]

#Q. Consider the version of the Dining-Philosophers problem in which the chopsticks are placed at the center of the table and any two of them can be used by a philosopher. Assume that requests for chopsticks are made one at a time. Describe a simple rule for determining whether a particular request can be satisfied without causing deadlock given the current allocation of chopsticks to philosophers.

"Do not grant the request, if there is no other Philosopher with 2 Chopsticks and if there is only Chopstick remaining"

[NAT]

#Q. A single-lane bridge connects the two Vermont villages of North Tunbridge and South Tunbridge. Farmers in the two villages use this bridge to deliver their produce to the neighboring town. The bridge can become deadlocked if a northbound and a southbound farmer get on the bridge at the same time. (Vermont farmers are stubborn and are unable to back up.) Using semaphores and/or mutex locks, design an algorithm in pseudocode that prevents deadlock. Initially, do not be concerned about starvation (the situation in which northbound farmers prevent southbound farmers from using the bridge, or vice versa).


```
BSEM mutex=1;
Acquire-Bridge( )
{
    P(mutex);
}
<Use_Bridge>
Release-Bridge()
{
    V(mutex);
}
```

[NAT]

#Q. Consider a logical address space of 64 pages of 1,024 words each, mapped onto a physical memory of 32 frames.

(a) How many bits are there in the logical address? 16 bits

(b) How many bits are there in the physical address? 15 bits

$N = 64$; $P.S = 1024$; $M = 32$

$L.A.S = 64 \times 1KW = 64KW$

$L.A = 16$ bits

$P.A.S = 32KW$

$P.A = 15$ bits

[NAT]

1024 By

#Q. Assuming a 1-KB page size, what are the page numbers and offsets for the following address references (provided as decimal numbers):

(a) 3085 (b) 42095

(c) 215201 (d) 650000

(e) 2000001

P = V.A / P.S

d = V.A % PS

V.A = 3085

P = 3085 / 1024 = 3 ✓

d = 308 % 1024 = 13 ✓

[NAT]

#Q. The BTV operating system has a 21-bit virtual address, yet on certain embedded devices, it has only a 16-bit physical address. It also has a 2-KB page size. How many entries are there in each of the following?

(a) A conventional, single-level page table : $2^{21}/2^{11} = 2^{10} = 1024$ ✓

(b) An inverted page table $\rightarrow 2^{16}/2^{11} = 2^{5} = 32$ ✓

[NAT]

#Q. Consider a paging system with the page table stored in memory.

(a) If a memory reference takes 50 nanoseconds, how long does a paged memory reference take? $(2m) = \underline{100\,ns}$

(b) If we add TLBs, and 75 percent of all page-table references are found in the TLBs, what is the effective memory reference time? (Assume that finding a page-table entry in the TLBs takes 2 nanoseconds, if the entry is present.)

$$EMAT_{TLB} = x(c+m) + (1-x)(c+2m)$$

$$c \sim 0$$

$$= 0.75(2+50) + 0.25(2+100)$$

$$= \underline{64.5}$$

[NAT]

#Q. Consider the following segment table:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?

219+430

(a) 0,430 (b) 1,10 ⇒ 2300+10

(c) 2,500 × (d) 3,400 ⇒ 1327+400

(e) 4,112 ×

[MCQ]

<Deadlock + MM>

#Q. Consider a system in which the total available memory is 48K and in which memory once allocated to a process cannot be preempted from that process. Three processes A, B, and C have declared in advance that the maximum amount of memory that they will require is 25K, 15K, and 41K words respectively. When the three processes are all in execution and using 3K, 9K, and 24K words of memory respectively, which one of the following requests for additional allocation can be granted with a guarantee that deadlock will not occur as a result of the allocation.

A. A requests 9K words.

B. C requests 7K words.

C. B requests 6K words.

D. A requests 6K words.

Total = 48K

	Max	Alloc	Need	Avail
A	25K	3K	22K	12K
B	15K	9K	6K	
C	41K	24K	17K	
	Total	36K		

#Q. Suppose the page table for the process currently executing on the processor looks like the following. All numbers are decimal, everything is numbered starting from zero, and all addresses are memory byte addresses. The page size is 1,024 bytes.

Virtual page number	Valid bit	Reference bit	Modify bit	Page frame number
0	1	1	0	4
1	1	1	1	7
2	0	0	0	-
3	1	0	0	2
4	0	0	0	-
5	1	0	1	0

a. Describe exactly how, in general, a virtual address generated by the CPU is translated into a physical main memory address.

b. What physical address, if any, would each of the following virtual addresses correspond to? (Do not try to handle any page faults, if any.)

(i) 1,052
(ii) 2,221
(iii) 5,499

V.A = $\boxed{P|d}$

(i) 1,052

P, d

P.A = $(7 * 1024) + 52$

$= 7168$
52
7220

$\frac{7220}{1024} = 7$ f

$d = 52$

P.A: $\boxed{7|52}$ (f, d)

#Q. Assuming a page size of 4 Kbytes and that a page table entry takes 4 bytes, how many levels of page tables would be required to map a 64-bit address space, if the top-level page table fits into a single page?

Condition for O.P.T Size to fit in one Page-frame

$V.A.S = 2^{S}$ By $\Rightarrow 2^{64}$

$P.S = 2^{x}$ By $\Rightarrow 2^{12}$

$PTE = 2^{c}$ Bytes $\Rightarrow 2^{2}$

No. of levels of Paging = 'l'

$$\Rightarrow \left[2^{S - l.x + l.c}\right] = 2^{x}$$

$$S - l.x + l.c = x$$

$$\boxed{64 - l.12 + l.2 = 12}$$

$\therefore l = 6$

#Q. Assume in a computer supporting paging, virtual addresses are 36 bits long. Page size is 16 KB. Assume a page table entry is 8 bytes long. The computer has 4 GB of RAM (physical memory).
We have three processes (A, B, C) in the system at the moment. The virtual memory (VM) layouts of the processes are like the following :

- A is using 64 MB from the start of its VM and 32 MB from the end. The rest of its VM is unused.
- B is using 128 MB from the start and 64 MB from the end. The rest of its VM is unused.
- C is using 32 MB from the start and 32 MB from the end. The rest of its VM is unused.

What will be the amount of physical memory required (in K Bytes) to hold the page table information for these three processes in total, if two-level page table is used with address division scheme (11,11,14)?

P.S = 16KB; V.A = 36 bits, P.T.E = 8B ; PAS = 4GB

Process "A"

16KB

$\left\{ 16KB + 3 * 16KB \right\}$

4 * 16KB

= 64KB ✓

8B

0 ✓

1 ✓

n ✓

2K

11 bits

OPT

2 Pages

4K entries

1_level P.T

8B

P_0

P_1

P_m

4m entries

11

2K entries chunk

2K

P.T = 32MB

$V.A.S = \frac{2^{36}}{2^{12}} = 2^{22}$ = 4m

64MB

$N = \frac{64MB}{16KB} = \frac{2^{26}}{2^{14}} = 2^{12}$ = 4K

$\frac{32MB}{16KB} = \frac{2^{25}}{2^{14}} = 2^{11}$ = 2K

#Q. Assume that a task is divided into four equal-sized segments and that the system builds an eight-entry page descriptor table for each segment. Thus, the system has a combination of segmentation and paging. Assume also that the page size is 2 Kbytes.

H/W

1. What is the maximum size of each segment?

2. What is the maximum logical address space for the task?

3. Assume that an element in physical location 00021 ABC is accessed by this task. What is the format of the logical address that the task generates for it? What is the maximum physical address space for the system?

#Q. A Virtual Memory has a page size IK words. There are eight pages and four blocks. The associative memory page table contains the following entries:

H/W

Page	Block (frame)
0	3
1	1
4	2
6	0

Make a list of all virtual addresses (in decimal) that will cause a page fault if used by the CPU.

#Q. A virtual memory system has an address space of 8K words, a memory space of 4K words, and page and block sizes of IK words. The following page reference changes occur during a given time interval. (Only page changes are listed. If the same page is referenced again, it is not listed twice.)

4 2 0 1 2 6 1 4 0 1 0 2 3 5 7

Determine the four pages that are resident in main memory after each page reference change if the replacement algorithm used is

(a) FIFO (c) Optimal

(b) LRU

[MCQ]

#Q. In partition-memory management scheme, Internal Fragmentation occurs when:

A Sufficient memory is available to run a program, but it is scattered between existing partitions.

B Insufficient memory is available to run a program.

C The partition allocated to a program is larger than the memory required by the program.

D A program is larger than the size of memory on the computer.

[MCQ]

#Q. The FIFO page-replacement policy:

A. Is based on program locality.

B. Sometimes can cause more page faults when memory size is increased.

C. Is not easy to implement, and hence, most systems use an approximation of FIFO.

[MSQ]

<A + C>

#Q. Which of the following statements is/are FALSE?

A. With the Least Recently Used (LRU) page-replacement policy, when the page size is halved, the number of page faults can be more than double the original number of page faults.

F

increasing

B. The working set size is a monotonically nondecreasing function of the working set parameter.

T

C. When the working set policy is used, main memory can never contain pages which do not belong to the working set of any program.

F

[MSQ]

#Q. It is advantageous for the page size to be large because:

A. Less unreferenced data will be loaded into memory.

B. Virtual addresses will be smaller.

C. Page tables will be smaller.

D. Leads to less page faults.

[MSQ]

#Q. It is advantageous for the page size to be small because:

A. Less unreferenced data will be loaded into memory.

B. Virtual addresses will be smaller.

C. Page tables will be smaller.

D. Leads to Less Internal Fragmentation.

[MCQ]

#Q. For a certain page trace starting with no page in the memory, a demand-paged memory system operated under the LRU replacement policy results in 9 and 11 page faults when the primary memory is of 6 and 4 pages, respectively. When the same page trace is operated under the optimal policy, the number of page faults may be:

A. 9 and 7.

B. 7 and 9.

C. 10 and 12.

D. 6 and 7.

L.R.U

(More than 6 Unique Pages)

7

6 Frames → 9 PF's

4 Frames → 11 PF's

Optimal

6 Frames →

4 Frames →

[MCQ]

#Q. In a Paged Segmented scheme of memory management, the segment table points to a page table because:

A The segment table may occasionally be too large to fit in one page.

B Each segment may be spread over a number of pages.

C Page size is usually larger than the segment size.

D The page table may be too large to fit into a single segment.

[MSQ]

#Q. Sharing in a paged memory system is done by:

A. Giving a copy of the shared pages to each process.

B. Dividing the program into procedures and data and allowing only the procedures to be shared.

C. Several page table entries pointing to the same frame in the main memory.

[MSQ]

#Q. Which of the following is/are FALSE with regard to sharing segments in a segmented system.

A. Maintaining a common segment table containing the information about shared segments.

B. Dividing the program into procedures and data and allowing only the procedures to be shared.

C. Dividing the shared segment into a set of pages and allowing only certain pages to be shared.

D. Sharing of Segments is never possible.

[MSQ]

#Q. Advantage of Dynamic Linking is/are that:

A. The segments that are not used in a run need not be linked into the process address space.

B. It reduces execution time overhead.

C. Only one copy of the Library can be maintained in Memory.

D. The linker need not construct the known segment table.

[NAT]

#Q. Assume that you have a page-reference string for a process with single frame (initially empty). The page-reference string has length p, and n distinct page numbers occur in it. Answer these questions for any page-replacement algorithms:

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

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

[NAT]

#Q. 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 one, two, three, four, five, six, and seven frames? Remember that all frames are initially empty, so your first unique pages will cost one fault each.

- LRU replacement
- FIFO replacement
- Optimal replacement

[MSQ]

#Q. Which is not relevant with regard to Disk scheduling :

A. Allocating disk space to users in a fair manner.

B. Validating the file control information stored in the file.

C. Examining pending disk requests to determine the most efficient way to service the requests.

D. Reorganizing disk requests to minimize seek time.

[MSQ]

#Q. Which of the following is normally contained in the directory entry of a file?

<A + B + D>

A. Creation date.

B. Access control list.

C. A count of the number of free blocks in the disk.

D. Filename and extension.

[MCQ]

#Q. Which of the following is an example of a spooled device?

A. A line printer used to print the output of a number of jobs.

B. The terminal used to enter the input data for a Fortran program being executed.

C. The secondary memory device in a virtual memory system.

D. The swapping area on a disk used by the swapper.

#Q. A UNIX file system has 1-KB blocks and 4-byte disk addresses. What is the maximum file size if i-nodes contain 10 direct entries, and one single, double, and triple indirect entry each?

$DBS = 1KB$; $DBA = 4B$; $N = \frac{1KB}{4B} = \underline{\underline{256}}$.

$File\ Size = 10 \times 1KB + 256 * 1KB + 256 * 256 * 1KB = 2^{26} = 64MB$

$+ 256 * 256 * 256 * 1KB$

$2^8 \times 2^8 \times 2^8 \times 2^{10} = 2^{34} = \underline{16GB}$

$\Rightarrow 16.064\ GB$

#Q. In UNIX System V, the length of a block is 1 Kbyte, and each block can hold a total of 256 block addresses. Using the i-node scheme, what is the maximum size of a file?

D. + S.I + D.I

[NAT]

#Q. Consider a file system that uses I-nodes to represent files. Disk blocks are 8 KB in size, and a pointer to a disk block requires 4 bytes. This file system has 12 direct disk blocks, as well as single, double, and triple indirect disk blocks. What is the maximum size of a file that can be stored in this file system?

#Q. Consider a file system that uses i-nodes to represent files. Disk blocks are 8 KB in size, and a pointer to a disk block requires 4 bytes. This file system has 12 direct disk blocks, as well as single, double, and triple indirect disk blocks. What is the maximum size of a file that can be stored in this file system?

#Q. Ignoring overhead for directories and file descriptors, consider a file system in which files are stored in blocks of 16K. bytes. For each of the following file sizes, calculate the percentage of wasted file space due to incomplete filling of the last block: 41,600 bytes; 640,000 bytes; 4,064,000 bytes.

D.B.S = 16KB

1) File Size = 41,600B $\Rightarrow N_{Blocks} =$

[NAT]

Disk Scheduling

#Q. Suppose that a disk drive has 5,000 cylinders, numbered 0 to 4,999. The drive is currently serving a request at cylinder 2,150, and the previous request was at cylinder 1,805. The queue of pending requests, in FIFO order, is:

2,069, 1,212, 2,296, 2,800, 544, 1,618, 356, 1,523, 4,965, 3681

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

a. FCFS

b. SSTF

c. SCAN

d. LOOK

e. C-SCAN

f. C-LOOK

2 mins Summary

Topic One

Topic Two

Topic Three

Topic Four

Topic Five

Disk Scheduling

THANK - YOU