

T3172

April/May 2008

CS1252-Operating Systems

Staff Incharge: T.Manonmani

Part A –(10*2= 20 marks)

1. Specify the critical factor strictly followed in real-time systems.
2. List out the three main advantages of multiprocessor system.
3. What is co-operating process?
4. What is bounded waiting in critical region?
5. What are the four necessary conditions a system should possess in order to be termed as deadlock?
6. What is segmentation?
7. Why should we use virtual memory?
8. What is meant by Thrashing?
9. Name the entries that make up a File Control Block(FCB).
10. State any two distinguishing features of UNIX and Windows.

Part B-(5*16= 80 marks)

11. a) i) Explain the facilities provided by the following operating systems.(10)
 - i) Clustered system and
 - ii) Real time System
- ii) List out the services provided by operating systems to programs and to the users of program.(6)

Or

- b) i) Explain the process creation and process termination process on process.(8)
 - ii) Write short notes on co-operating process and schedulers. (8)
-
12. a) Describe the following scheduling algorithms(16)
 - Shortest-job-first scheduling
 - Round –robin scheduling
 - Real-time scheduling
 - Priority scheduling

Or

b) What is the important feature of critical section? State the dining philosophers problem and show how to allocate the several resources among several processes in a deadlock and starvation free manner. (16)

13. a) i) How can deadlock be detected? Explain. (10)

ii) Write short notes on swapping (6)

Or

b) Discuss the advantages of paging memory management and the conversion of logical address into physical address with necessary hardware (16)

14. a) Discuss the following page replacement algorithms, giving a suitable page reference string i) LRU ii) FIFO and iii) Optimal (5+5+6)

Or

b) i) State the various attributes of a file and their purpose. Discuss the various file operations. (10)

ii) Discuss about demand paging. (6)

15. a) Describe in detail any three methods of implementing the file system (16)

Or

b) Write short notes on:

a. Disk structure (5)

b. Indexed allocation (5)

c. Shortest-Seek-Time-First (SSTF) scheduling. (6)

T3172**April/May 2008****CS1252-Operating Systems****Part A –(10*2= 20 marks)**

~~1. Specify the critical factor strictly followed in real-time systems.~~

~~2. List out the three main advantages of multiprocessor system.~~

3. What is co-operating process?

A process is co-operating if it can affect or be affected by the other processes executing in the system. Any process that share data with other process is a co-operating process.

4. What is bounded waiting in critical region?

Each process has a segment of code called a critical section, in which the process may be changing common variables, updating a table, writing a file, and so on.

5. What are the four necessary conditions a system should posses in order to be termed as deadlock?

- Mutual exclusion.
- Hold and wait
- No preemption
- Circular wait.

6. What is segmentation?

Segmentation is a memory management that supports the user view of memory.

A particular advantage of segmentation is the association of protection bits with the segments to prevent illegal accesses to memory. Another advantage involves the sharing of code or data.

7. Why should we use virtual memory?

When the program's memory size is larger than available physical memory size then , we need virtual memory. Here the job or process to be executed that may not be completely in primary memory, but partial portion of the process or job may be in secondary memory. It requires CPU scheduling scheme , job synchronization , job communication also jobs should not get stuck in deadlock.

8. What is meant by Thrashing?

High paging activity is called thrashing. A process is thrashing if it is spending more time paging than executing. The page fault rate increases tremendously during thrashing.

9. Name the entries makes up a File Control Block (FCB).

file permissions
file dates (create, access, write)
file owner, group, ACL
file size
file data blocks

10. State any two distinguishing features of UNIX and Windows.

- With a command-line operating system (e.g., DOS), you type a text command and the computer responds according to that command.
- With a graphical user interface (GUI) operating system (e.g., Windows), you interact with the computer through a graphical interface with pictures and buttons by using the mouse and keyboard.
- With Unix you have in general the option of using either command-lines (more control and flexibility) or GUIs (easier).

Part B-(5*16= 80 marks)

~~11. a) i) Explain the facilities provided by the following operating systems.(10)~~

- ~~i) ——— Clustered system and
ii) ——— Real time System~~

ii) List out the services provided by operating systems to programs and to the users of program.(6)

- Program execution
- I/O operations
- File-system manipulation
- Communications
- Error detection
- Resource allocation
- Accounting
- Protection

Or

b) i) Explain the process creation and process termination process on process.(8)

A program loaded into memory and executing is commonly referred to as process.

Process Creation:

- A new process is created using the fork System call.
- The parent continues to execute concurrently with its children.
- Parent waits until some or all its children have terminated.

- Child process is a duplicate of the parent process.
- Child process has a program loaded into it.

Process termination:

- Process terminates using the exit system call.

A parent terminates the execution of its child for the reasons such as:

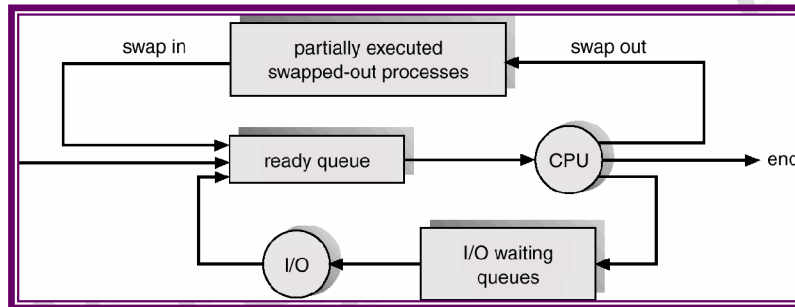
- The child has exceeded the usage of its resources.
- Task assigned to the child is no longer required.
- If the parent process exits.

ii) Write short notes on co-operating process and schedulers. (8)

A process is cooperating if it can affect or affected by the other processes executing in the system. Clearly, any process that shares data with other processes is a cooperating process.

The operating system must select one of the processes in the ready queue to be executed. The selection process is carried out by the short term scheduler or CPU scheduler.

- Long-term scheduler (or job scheduler) – selects which processes should be brought into the ready queue.
- Short-term scheduler (or CPU scheduler) – selects which process should be executed next and allocates CPU.



- Short-term scheduler is invoked very frequently (milliseconds) \Rightarrow (must be fast).
- Long-term scheduler is invoked very infrequently (seconds, minutes) \Rightarrow (may be slow).
- The long-term scheduler controls the *degree of multiprogramming*.
- Processes can be described as either:
 - *I/O-bound process* – spends more time doing I/O than computations, many short CPU bursts.
 - *CPU-bound process* – spends more time doing computations; few very long CPU bursts.

12. a) Describe the following scheduling algorithms(16)

Shortest-job-first scheduling

- Associate with each process the length of its next CPU burst. Use these lengths to schedule the process with the shortest time.
- Two schemes:
 - nonpreemptive – once CPU given to the process it cannot be preempted until completes its CPU burst.
 - preemptive – if a new process arrives with CPU burst length less than remaining time of current executing

process, preempt. This scheme is known as the Shortest-Remaining-Time-First (SRTF).

- SJF is optimal – gives minimum average waiting time for a given set of processes.

Round –Robin scheduling

- Each process gets a small unit of CPU time (*time quantum*), usually 10-100 milliseconds. After this time has elapsed, the process is preempted and added to the end of the ready queue.
- If there are n processes in the ready queue and the time quantum is q , then each process gets $1/n$ of the CPU time in chunks of at most q time units at once. No process waits more than $(n-1)q$ time units.
- Performance
- q large \Rightarrow FIFO
- q small $\Rightarrow q$ must be large with respect to context switch, otherwise overhead is too high.

Real-time scheduling

- *Hard real-time* systems – required to complete a critical task within a guaranteed amount of time.
- *Soft real-time* computing – requires that critical processes receive priority over less fortunate ones.

Priority scheduling

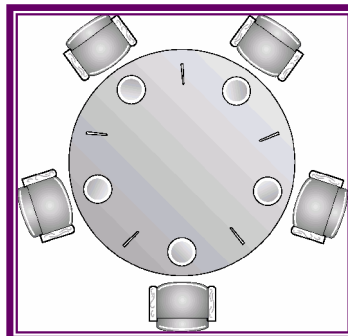
- A priority number (integer) is associated with each process
- The CPU is allocated to the process with the highest priority (smallest integer \equiv highest priority).
 - Preemptive
 - Non-preemptive
- SJF is a priority scheduling where priority is the predicted next CPU burst time.
- Problem \equiv Starvation – low priority processes may never execute.
- Solution \equiv Aging – as time progresses increase the priority of the process.

Or

b) What is the important feature of critical section? State the dining philosophers problem and show how to allocate the several resources among several processes in a deadlock and starvation free manner. (16)

Each process has a segment of code called a critical section, in which the process may be changing common variables, updating a table, writing a file, and so on.

Dining-Philosophers Problem



- Shared data
 - **semaphore chopstick[5];**
- Initially all values are 1
- Philosopher i :


```
do {
    wait(chopstick[i])
    wait(chopstick[(i+1) % 5])
    ...
    eat
    ...
    signal(chopstick[i]);
    signal(chopstick[(i+1) % 5]);
    ...
    think
    ...
} while (1);
```

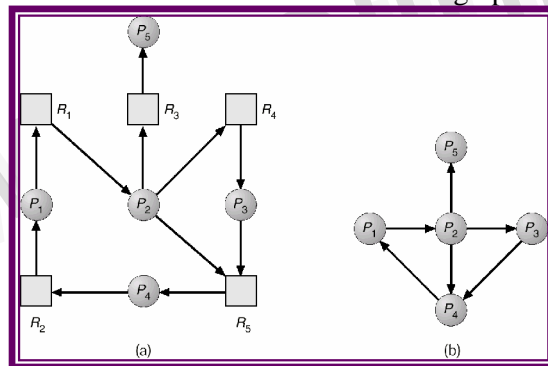
13. a) i) How can deadlock be detected? Explain. (10)

Allow system to enter deadlock state

Detection algorithm

Recovery scheme

- Maintain *wait-for* graph
 - Nodes are processes.
 - $P_i \rightarrow P_j$ if P_i is waiting for P_j .
- Periodically invoke an algorithm that searches for a cycle in the graph.
- An algorithm to detect a cycle in a graph requires an order of n^2 operations, where n is the number of vertices in the graph



Several Instances of a Resource Type

- **Available:** A vector of length m indicates the number of available resources of each type.
- **Allocation:** An $n \times m$ matrix defines the number of resources of each type currently allocated to each process.
- **Request:** An $n \times m$ matrix indicates the current request of each process. If $Request[ij] = k$, then process P_i is requesting k more instances of resource type R_j .

Detection Algorithm

Let *Work* and *Finish* be vectors of length m and n , respectively Initialize:

(a) $Work = Available$

- (b) For $i = 1, 2, \dots, n$, if $Allocation_i \neq 0$, then
 $Finish[i] = false$; otherwise, $Finish[i] = true$.
2. Find an index i such that both:
- (a) $Finish[i] == false$
- (b) $Request_i \leq Work$
- If no such i exists, go to step 4.
 $Work = Work + Allocation_i$
 $Finish[i] = true$
 go to step 2.
4. If $Finish[i] == false$, for some i , $1 \leq i \leq n$, then the system is in deadlock state. Moreover, if $Finish[i] == false$, then P_i is deadlocked.

ii) Write short notes on swapping (6)

A process needs to be in memory to be executed. However a process can be swapped temporarily out of memory to a backing store and then brought back into memory for continued execution.

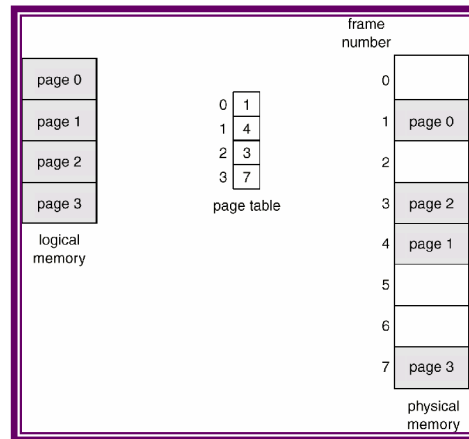
This process is called swapping.

- A process can be *swapped* temporarily out of memory to a *backing store*, and then brought back into memory for continued execution.
- Backing store – fast disk large enough to accommodate copies of all memory images for all users; must provide direct access to these memory images.
- *Roll out, roll in* – swapping variant used for priority-based scheduling algorithms; lower-priority process is swapped out so higher-priority process can be loaded and executed.
- Major part of swap time is transfer time; total transfer time is directly proportional to the *amount* of memory swapped.
- Modified versions of swapping are found on many systems, i.e., UNIX, Linux, and Windows.

Or

b) Discuss the advantages of paging memory management and the conversion of logical address into physical address with necessary hardware (16)

- Logical address space of a process can be noncontiguous; process is allocated physical memory whenever the latter is available.
- Divide physical memory into fixed-sized blocks called **frames** (size is power of 2, between 512 bytes and 8192 bytes).
- Divide logical memory into blocks of same size called **pages**.
- Keep track of all free frames.
- To run a program of size n pages, need to find n free frames and load program.
- Set up a page table to translate logical to physical addresses.
- Internal fragmentation.
- Address generated by CPU is divided into:
 - *Page number (p)* – used as an index into a *page table* which contains base address of each page in physical memory.
 - *Page offset (d)* – combined with base address to define the physical memory address that is sent to the memory unit.



- Page table is kept in main memory.
- *Page-table base register* (PTBR) points to the page table.
- *Page-table length register* (PRLR) indicates size of the page table.
- In this scheme every data/instruction access requires two memory accesses. One for the page table and one for the data/instruction.
- The two memory access problem can be solved by the use of a special fast-lookup hardware cache called *associative memory* or *translation look-aside buffers* (TLBs)

14 a) Discuss the following page replacement algorithms, giving a suitable page reference string i) LRU ii) FIFO and iii) Optimal (5+5+6)

- Prevent over-allocation of memory by modifying page-fault service routine to include page replacement.
- Use *modify (dirty) bit* to reduce overhead of page transfers – only modified pages are written to disk.
- Page replacement completes separation between logical memory and physical memory – large virtual memory can be provided on a smaller physical memory.

In **Optimal algorithm**, it replaces the page that will not be used for the longest period of time. Optimal algorithm has the lowest page-fault rate of all algorithms.

Even though optimal algorithm has the smallest page fault-rate, it is difficult to implement, because it requires future knowledge of the reference string. As a result, it is used mainly for comparison studies.

In Least Recently Used algorithm, it replaces the page that has not been used for the longest period of time. LRU replacement associates with each page the time of that page's last use. It can be implemented using counters or stack.

Belady's anomaly reflects the fact that, for some page-replacement algorithms, the page-fault rate may increase as the number of allocated frame increases.

A **FIFO** page replacement algorithm associates with each page the time when that page was brought into memory. When a page must be replaced, the oldest page is chosen. It is not strictly necessary to record the time when a page is brought in.

Or

b) i) State the various attributes of a file and their purpose. Discuss the various file operations. (10)

File attributes:

- Name – only information kept in human-readable form
- Type – needed for systems that support different types
- Location – pointer to file location on device
- Size – current file size
- Protection – controls who can do reading, writing, executing
- Time, date, and user identification – data for protection, security, and usage monitoring
- Information about files are kept in the directory structure, which is maintained on the disk

Operations

- Create
- Write
- Read
- file seek – reposition within file
- Delete
- Truncate
- Open(F_i) – search the directory structure on disk for entry F_i , and move the content of entry to memory
- Close (F_i) – move the content of entry F_i in memory to directory structure on disk

ii) Discuss about demand paging. (6)

- Page Fault Rate $0 \leq p \leq 1.0$
 - if $p = 0$ no page faults
 - if $p = 1$, every reference is a fault
- Effective Access Time (EAT)

$EAT = (1 - p) \times \text{memory access} + p (\text{page fault overhead} + [\text{swap page out}] + \text{swap page in} + \text{restart overhead})$

- Memory access time = 1 microsecond
- 50% of the time the page that is being replaced has been modified and therefore needs to be swapped out.
- Swap Page Time = 10 msec = 10,000 msec
 - $EAT = (1 - p) \times 1 + p (15000)$
 - $1 + 15000P$ (in msec)

15 a) Describe in detail any three methods of implementing the file system (16)

Refer Nov/Dec 2008 15 b) i)

Or

b) Write short notes on:

d. Disk structure (5)

- Disk drives are addressed as large 1-dimensional arrays of *logical blocks*, where the logical block is the smallest unit of transfer.
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially.
- Sector 0 is the first sector of the first track on the outermost cylinder.

- Mapping proceeds in order through that track, then the rest of the tracks in that cylinder, and then through the rest of the cylinders from outermost to innermost.

e. Indexed allocation (5)

Refer Nov/Dec 2008 15 b) i)

f. Shortest-seek-Time-First (SSTF) scheduling. (6)

- Selects the request with the minimum seek time from the current head position.
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests.
- Illustration shows total head movement of 236 cylinders.

