VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI

OPERATING SYSTEMS QUESTION BANK

(15CS64)

(As per Visvesvaraya Technological University Syllabus)

Compiled By:

Prof. Meghashri E M
(Assistant Professor, Dept. of CSE)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SAI VIDYA INSTITUTE OF TECHNOLOGY

(Affiliated to VTU, Belgaum, Approved by AICTE, New Delhi and Govt. of Karnataka)
Rajanukunte, Doddaballapur Road, Bangalore-560064
Tel: 080-2846 8196, Fax: 2846 8193 / 98, Web: www.saividya.ac.in

2016-17

Disclaimer

The information contained in this document is the proprietary and exclusive property of Sai Vidya Institute of Technology except as otherwise indicated. No part of this document, in whole or in part, may be reproduced, stored, transmitted, or used for course material development purposes without the prior written permission of Sai Vidya Institute of Technology.

The information contained in this document is subject to change without notice. The information in this document is provided for informational purposes only.

Trademark

Edition: 2017 - 18

Document Owner

The primary contacts for questions regarding this document are:

Author : Prof. Meghashri E M

Department: Computer Science & Engineering

Contact email id :

meghashree.em@saividya.ac.in

OPERATING SYSTEMS

Subject Code: 15CS64

Syllabus

Module–1 **10 Hours**

Introduction to Operating Systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing Environments. Operating System Services; User – Operating System interface; System calls; Types of system calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot. **Process Management**: Process concept; Process scheduling; Operations on processes; Inter-process communication.

Module-2 **10 Hours**

Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. **Process Scheduling:** Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling. **Process Synchronization**: Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Module-3 **10 Hours**

Deadlocks: Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Memory Management:** Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.

Module-4 **10 Hours**

Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System**, Implementation of File System: File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; **File system implementation**; Directory implementation; Allocation methods; Free space management

Module-5 **10 Hours**

Secondary Storage Structures, Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems. **Case Study:** The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication.

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009. (Chapters: 1, 2, 3.1 to 3.4 , 4.1 to 4.4, 5.1 to 5.5, 6.1 to 6.7, 7, 8.1 to 8.6, 9.1 to 9.6, 10, 11.1 to 11.5, 12.1 to 12.6, 17.1 to 17.8, 21.1 to 21.9)

Reference Books:

1. D.M Dhamdhere: Operating systems - A concept based Approach, 2nd Edition, Tata McGraw-Hill, 2002.
2. P.C.P. Bhatt: Introduction to Operating Systems: Concepts and Practice, 2nd Edition, PHI, 2008.
3. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 1990.

Introduction to Operating Systems, System structures

June/July 2017

1. What is OS? Explain multiprogramming and time sharing systems. (6)
2. Explain dual mode operation in OS with a neat block diagram (4)
3. What are system calls? Briefly point out its types. (4)
4. What are virtual machines? Explain with block diagram. Point out its benefits (6)

May 2017

1. Explain the advantages of layered approach, with a diagram. (6)
2. Write the system call sequence to copy a file from source to destination (7)
3. With a neat diagram, explain the concept of virtual machines (7)

Dec-2015

1) Define an Operating System? What is system's viewpoint of an Operating System? Explain the dual mode operation of an Operating System. (8)
2) Explain the types of multiprocessor systems and the types of clustering. What are fault tolerant systems? (6)
3) Explain the concept of virtual machines. (6)

June-2015

4) What are the activities for which the operating system is responsible for, in connection with:
   i) Process management ii) File management (10)
5) Explain any two types of system calls. (5)
6) What are virtual machines? Explain the benefit of creating virtual machines. (5)

Dec - 2014

7) Differentiate between multiprogramming and multiprocessing. (5)
8) What are the different ways in which the Pthread terminates. (5)
9) Explain any two facilities provided for implementing interacting process in programming language and operating system. (5)

Dec-2013

10) Explain the advantages of the **layered approach** with a neat diagram (6)
11) What are the essential properties of **batch, real time and distributed operating systems** (6)

June-July 2013

12) Is separation of mechanism and policy desirable while designing an operating system? Discuss with example. (8)
13) With a neat diagram of VM-Ware architecture, explain the concept of VM and the main advantage of using VM architecture. (8)

Dec-2012

14) Define **Operating System**? Discuss its role with respect to user and system viewpoint. (6)
15) Explain how an Operating System can be viewed as a resource manager. (4)
16) What is a **distributed operating system**? What are the advantages of the distributed operating syste (6)
17) What are **system calls** ? With examples explain different categories of system calls (7)
18) Briefly explain the **clustered systems** and real time systems.(4)

Dec-2011

19) Explain the '**graceful degradation'** and '**fault tolerant'** in a multiprocessor system (6)
20) What is a 'virtual machine'? Explain the just-in-time (JIT) compiler, used in a java virtual machine (6)

May-2010

21) Write and explain the sequence of system calls for copying a file to another (new) file.(5)
22) Define: (i) Micro Kernel (ii) Bootstrap program (iii) Caching (iv) trap (v) Job Pool (10)
23) Explain two sets of OS Services that are helpful to user as well as efficient operation of system. (5)

Dec-2010

24) What are the **OS operations**? Explain. (6)

Dec-2009

25) Give the features of **symmetric** and **asymmetric** multiprocessing systems. (4)
26) Briefly explain the **common classes of services** provided by the various operating systems for helping the user and for ensuring the efficient operation of the system. (10)

Dec-2008

27) Differentiate between a trap and an interrupt. (2)
28) List and explain the advantages of multi processor system.
29) Explain the I/O structure.

Process Management

June/July 2017

1. Why is it important for the scheduler to distinguish I/O bound programs from CPU bound programs? (2)
2. What is interprocess communication? Explain its types. (6)
3. Consider the following set of processes, with the length of the CPU burst given in milliseconds. (12)

Processes	Burst time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

The processes are assumed to have arrived in the order P1,P2,P3,P4,P5 all at time 0.

i) Draw the Gantt charts for the following scheduling algorithms, FCFS, SJF, RR(quantum=1)

ii) Find out turn around time & waiting time of each process for each of these scheduling algorithm and also find out average turn around time and average waiting time. (12)

May 2017

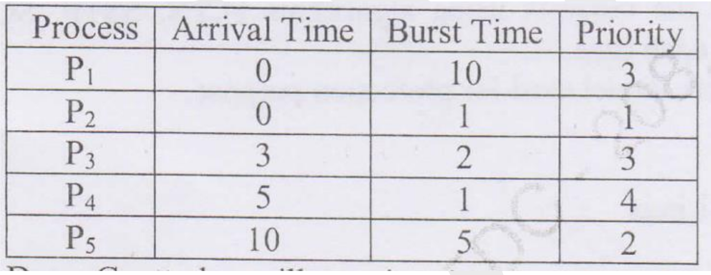
1. Explain the process states with diagram (6)
2. Explain the different multithreading models, with neat sketches. (6)
3. Consider the following set of processes. Draw Gantt charts and calculate average waiting time & average turn around time using non-preemptive SJF & preemptive SJF scheduling algorithms. (8)

Process	Arrival time	Burst time
P1	0	8
P2	1	4
P3	2	9
P4	3	5

Dec 2015

1) What is a process? Draw and explain the state diagram of a process. Give a note on context switch. (8)

2) Consider the following set of processes. Assume the length of CPU burst time is given in milliseconds.



Process	Arrival Time	Burst Time	Priority
P_1	0	10	3
P_2	0	1	1
P_3	3	2	3
P_4	5	1	4
P_5	10	5	2

Draw Gantt charts illustrating the execution of these processes using FCFS and pre-emptive priority scheduling algorithm. Assume highest priority = 1 and lowest priority = 4. Also, calculate average waiting time and average turn around time of both the algorithms. (6)

3) Discuss any three threading issues that come with multi-threaded programs. (6)

June 2015

4) Differentiate between direct and indirect inter process communication (4)

5) Explain the multithreading models (9)

6) Consider the following set of processes

Process	Arrival Time	Burst Time
P1	0	5
P2	1	1

P3	2	4

Compute average turn around time and average waiting time using FCFS, Preemptive SJF and RR(quantum-4)

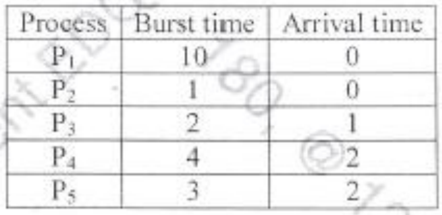
Dec 2014

7) Explain different scheduling criteria that must be kept in mind while choosing different scheduling algorithms.

Dec 2013

8) Differentiate between:
   a. Process and a thread
   b. Short term and medium term schedulers
   c. User level and kernel level threads
   d. Waiting and turn around time (8)

9) Consider the following set of processes with arrival time



Process	Burst time	Arrival time
P_1	10	0
P_2	1	0
P_3	2	1
P_4	4	2
P_5	3	2

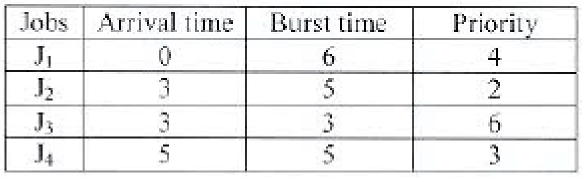
   i) Draw Gantt charts using FCFS, SJF preemptive and non-preemptive and RR scheduling (1 time unit)
   ii) Calculate the avg. Wating time for each of the scheduling algorithms (8)
   iii) Which of them provide minimal avg.turn around time and minimal avg. Waiting time.(
   iv) Find out the time at which there are maximum number of process in the waiting queue.(

10) Describe the actions an operating system takes to **context switch** between processes. (4)

Dec-2012

11) Explain the **process state transition** diagram. (6)

12) For the processes listed below. Draw Gantt charts using preemptive and non-preemptive priority scheduling algorithm. A larger priority number has a higher priority. (5)

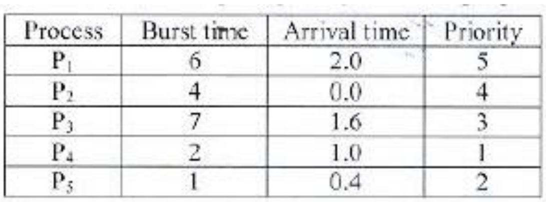


Jobs	Arrival time	Burst time	Priority
J_1	0	6	4
J_2	3	5	2
J_3	3	3	6
J_4	5	5	3

13) What is a **process**? With a state diagram, explain states of a process. Also write the structure of process control block (8)

14) Discuss the three common ways of establishing **relationship** between user and kernel threads. (6)

15) Is CPU scheduling necessary? For the following set of processes, find the average waiting time and average turn around time along with Gantt chart illustrations for –
i)Shortest remaining time first ii) Preemptive priority scheduling algorithms. (1- HIGH priority) (6)



Process	Burst time	Arrival time	Priority
P_1	6	2.0	5
P_2	4	0.0	4
P_3	7	1.6	3
P_4	2	1.0	1
P_5	1	0.4	2

<u>June-2012</u>

16) What is the **need** for context switching. (4)

17) What are the differences between **user level threads and kernel supported threads** (5)

<u>Dec-2011</u>

18) Explain the **benefits** of multithreaded programming (8)

<u>Dec-2010</u>

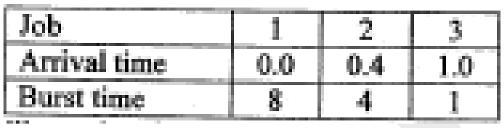
19) Define **IPC** (Inter Process Communication). What are the different methods used for logical implementation of a message passing system? Explain any one. (6)

20) Discuss three common ways of establishing relationship between the user thread and kernel thread. (6)

<u>Dec-2009</u>

21) Explain the various scheduling criteria (4)

22) Suppose the following jobs arrive for processing at the times indicated. Each job will run the listed amount of time



Job	1	2	3
Arrival time	0.0	0.4	1.0
Burst time	8	4	1

i) Give a Gantt chart illustrating the execution of these jobs, using the non pre-emptive FCFS and SJF scheduling algorithms.

ii) What is turnaround time and waiting time of each job for the above algorithms?

iii) Compute average turn around time if CPU is left idle for the first 1 unit and then SJF is used. (Job1 and Job2 will wait during this time) (10)

<u>Dec-2008</u>

23) Discuss the operations of process creation and process termination in UNIX (7)

24) Describe the implementation of IPC using shared memory and message passing.(8)

<u>June-2008</u>

25) Differentiate between long-term and short-term **schedulers**. (4)

26) What is process co-operation? Give reason for process co-operation(5)

27) Why is a thread called a LWP? (2)
28) Explain the different threading issues. (8)

Process Synchronization

June/July 2017

1. Define Semaphores. Explain its usage and implementation. (6)
2. What are monitors? Explain its usage and implementation. (8)
3. Explain Dining philosophers solution using monitors. (6)

May 2017

1. Explain the critical section problem. List and explain the requirement to be met by a solution to critical section problem. (8)
2. Describe the monitor solution to the classical dining –philosopher's problem. (8)
3. What do you mean by a binary semaphore and a counting semaphore? (4)

Dec 2015

1) What are semaphores? Explain any three use cases of semaphores. (4)
2) Describe an n-process solution to critical section problem which uses test and set () hardware instruction. Prove how its algorithm satisfies all the requirements of critical section problems solution. (8)
3) Discuss how Readers-Writers problem can be solved using semaphores. (8)

June-2015

4) Explain Peterson's solution to critical section problem (6)
5) Mention three classical problems of synchronization. Explain any one in detail. (8)

Dec-13

6) Explain **Dining-Philosopher's problem** using monitors. (10)
7) What is **race condition**? Explain Reader's writer's problem with semaphores (10)(Dec-13)

June-July 2013

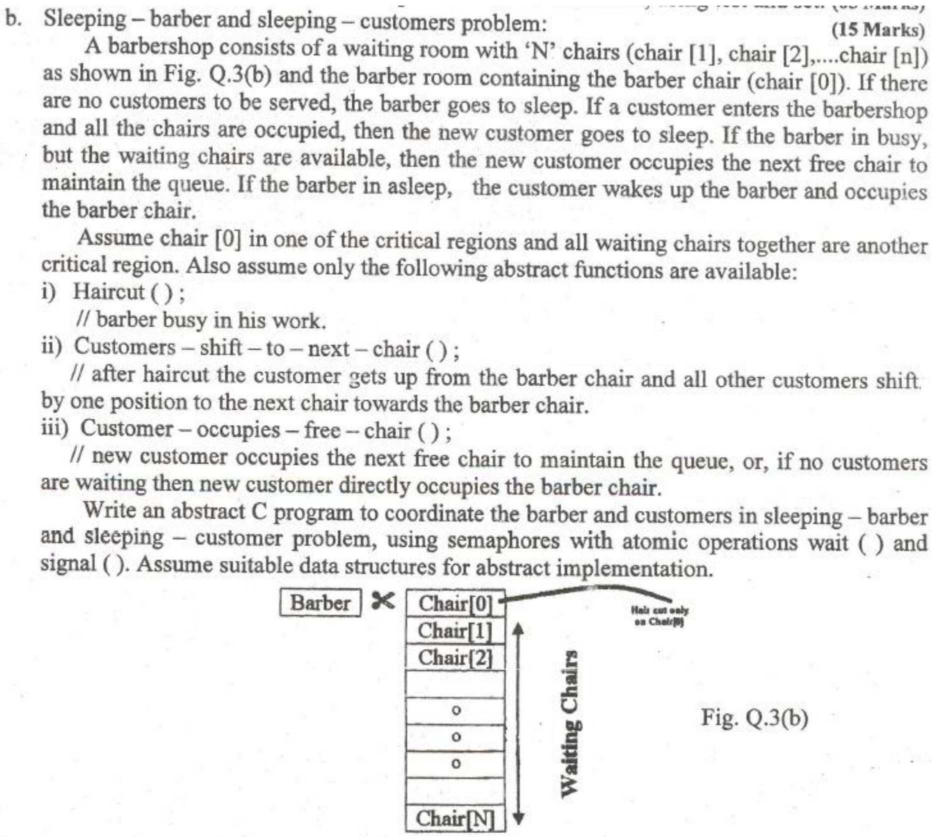
8) Explain the range of monitors with a schematic view of its structure; write a monitor solution to bounded-buffer problem. (8)
9) Describe an N-process solution to critical section problem which uses test and test() atomic instruction. Also explain how the algorithm satisfies all the requirements of critical section. (8)
10) Servers can be designed to limit the number of open connections. For example, a server may wish to have only N socket connections at any point in time. As soon as N connections are made, the server will not accept another incoming connection until an existing connection is released. Explain how semaphores can be used by a server to limit the number of concurrent connections. (4)

Dec-12

11) What is busy waiting in a **critical section** concept? How semaphore is used to solve critical section problem? What are the advantages of semaphore? (10)
12) What is **semaphore**? Explain how it can be used to solve the producer-consumer problem (7)
13) Define **mutual exclusion** and **critical section**. Write the software solution for 2- process synchronization. (7)

Dec-2011

14) Define **test and set** instruction and implement mutual exclusion, using test and set. (5)
15)



b. Sleeping – barber and sleeping – customers problem: **(15 Marks)**

A barbershop consists of a waiting room with 'N' chairs (chair [1], chair [2],....chair [n]) as shown in Fig. Q.3(b) and the barber room containing the barber chair (chair [0]). If there are no customers to be served, the barber goes to sleep. If a customer enters the barbershop and all the chairs are occupied, then the new customer goes to sleep. If the barber in busy, but the waiting chairs are available, then the new customer occupies the next free chair to maintain the queue. If the barber in asleep, the customer wakes up the barber and occupies the barber chair.

Assume chair [0] in one of the critical regions and all waiting chairs together are another critical region. Also assume only the following abstract functions are available:

i) Haircut ( ) ;
// barber busy in his work.

ii) Customers – shift – to – next – chair ( ) ;
// after haircut the customer gets up from the barber chair and all other customers shift by one position to the next chair towards the barber chair.

iii) Customer – occupies – free – chair ( ) ;
// new customer occupies the next free chair to maintain the queue, or, if no customers are waiting then new customer directly occupies the barber chair.

Write an abstract C program to coordinate the barber and customers in sleeping – barber and sleeping – customer problem, using semaphores with atomic operations wait ( ) and signal ( ). Assume suitable data structures for abstract implementation.

Fig. Q.3(b)

Dec 2010

16) What do you mean by binary semaphore and counting semaphore? Along with the necessary 'C'-struct, explain the implementation of wait() and signal() semaphore operations. (10)
17) Explain **monitor** with necessary syntax? (5)

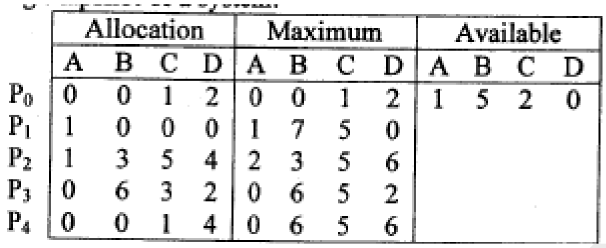
June – July 2009

18) Describe the Bounded-buffer problem and give a solution for the same using semaphores. Write the structure of producer and consumer processes. (8)
19) What are the three requirements to be met by a solution to the critical section problem? Explain. (6)

Deadlocks

June/July 2017

1. What are deadlocks? What are its characteristics? (5)
2. Consider the following snapshot of a system: (10)



	Allocation				Maximum				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	0	0	1	2	0	0	1	2	1	5	2	0
P_1	1	0	0	0	1	7	5	0				
P_2	1	3	5	4	2	3	5	6				
P_3	0	6	3	2	0	6	5	2				
P_4	0	0	1	4	0	6	5	6				

Answer the following questions using the Banker's algorithm:

i) What is the content of matrix need?

ii) Is the system in a safe state?

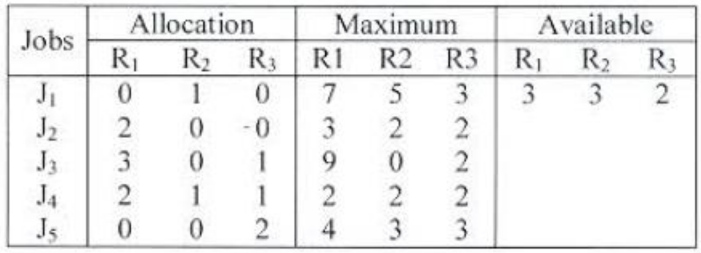
iii) If a request from process P1 arrives for(0,4,2,0) can the request be granted immediately?(10)

iv) Is the system in a safe state?

3 Explain the process of recovery from deadlock. (5)

May 2017

1) What are deadlocks? Explain the necessary conditions for its occurrence. (6)
2) System consists of five jobs (J1,J2,J3,J4,J5) and three resources (R1,R2,R3) . Resource type R1 has 10 instances, resource type R2 has 5 instances and R3 has 7 instances. The following snapshot of the system has been taken:



Jobs	Allocation			Maximum			Available		
	R_1	R_2	R_3	R1	R2	R3	R_1	R_2	R_3
J_1	0	1	0	7	5	3	3	3	2
J_2	2	0	-0	3	2	2			
J_3	3	0	1	9	0	2			
J_4	2	1	1	2	2	2			
J_5	0	0	2	4	3	3			

Find need matrix and calculate the safe sequence by using Banker's algorithm. Mention the above system is safe or not safe. (8)

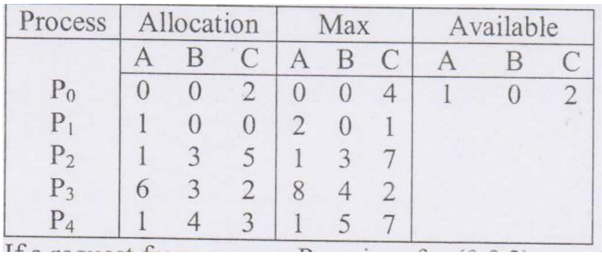
3) Describe RAG: (6)

i) With deadlock

ii) With a cycle but no deadlock

Dec 2015

1) With the help of a system model, explain a deadlock and explain the necessary conditions that must hold simultaneously in a system for a deadlock to occur. (6)
2) Using Banker's algorithm determine whether the system is in a safe state.



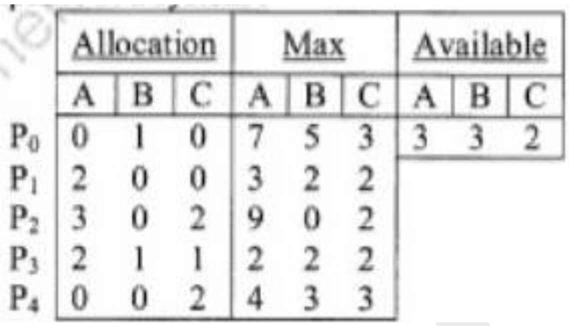
Process	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P_0	0	0	2	0	0	4	1	0	2
P_1	1	0	0	2	0	1			
P_2	1	3	5	1	3	7			
P_3	6	3	2	8	4	2			
P_4	1	4	3	1	5	7			

If a request from process P_2 arrives for (0 0 2), can the request be granted immediately? (10)

3) How is a system recovered from deadlock? (4)

Jun/July 2015

4) Consider the following snapshot of a system.



	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P_0	0	1	0	7	5	3	3	3	2
P_1	2	0	0	3	2	2			
P_2	3	0	2	9	0	2			
P_3	2	1	1	2	2	2			
P_4	0	0	2	4	3	3			

Answer the questions using Banker's algorithm.

i) What is the content of the matrix need?

ii) Is the system is safe state?

iii) If a request P1 arrives for (1,0,2), can the request be granted immediately? (12)

5) For the following resource-allocation graph write the corresponding wait-for graph.

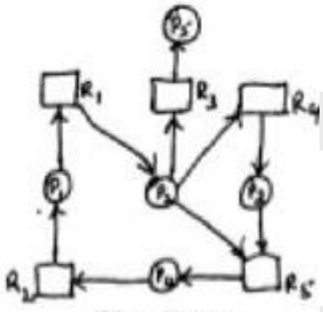


Fig. O4(b)

(4)

6) Explain the different methods used to recover from deadlock. (4)

Dec 2014

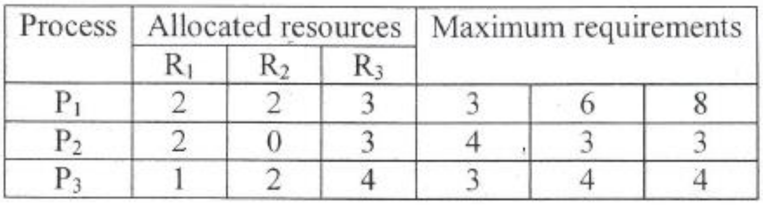
7) Consider a system containing m resources of the same type being shared by n processes. Resources can be requested and released by processes only one at a time. Show that the system is deadlock free if the following two conditions hold:

i) The maximum need of each process is between 1 and m resources.

ii) The sum of all maximum needs is less than m+n.

Dec-Jan 2013

8) For the following snapshot find the safe sequence using Banker's algorithm:
The number of resource units are R1,R2,R3 which are 7,7,10 respectively. (6)



Process	Allocated resources			Maximum requirements		
	R_1	R_2	R_3			
P_1	2	2	3	3	6	8
P_2	2	0	3	4	3	3
P_3	1	2	4	3	4	4

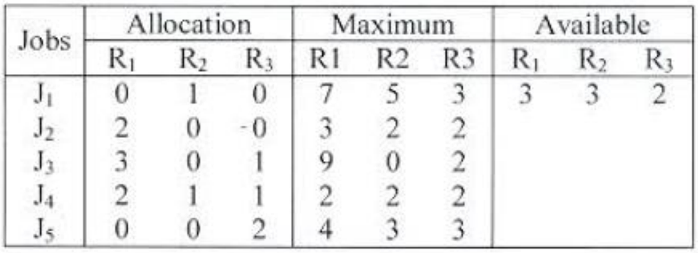
9) Explain different methods to **recover** from deadlock (6)

10) Dead lock exists if a cycle exists. Yes or no. Justify your answer with a suitable example. (8)

Dec-2012

11) What is **Resource Allocation Graph**(RAG)? Explain how RAG is very useful in describing deadly embrace (dead lock)by considering your own example. (8)

12) System consists of five jobs (J1,J2,J3,J4,J5) and three resources (R1,R2,R3) . Resource type R1 has 10 instances, resource type R2 has 5 instances and R3 has 7 instances. The following snapshot of the system has been taken:

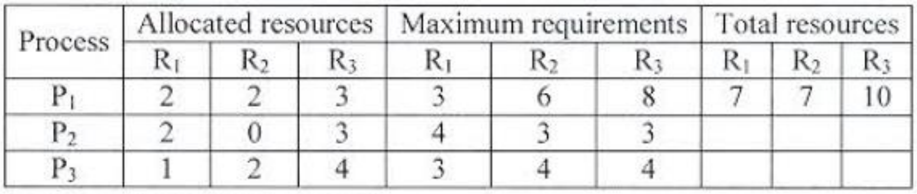


Jobs	Allocation			Maximum			Available		
	R_1	R_2	R_3	R1	R2	R3	R_1	R_2	R_3
J_1	0	1	0	7	5	3	3	3	2
J_2	2	0	-0	3	2	2			
J_3	3	0	1	9	0	2			
J_4	2	1	1	2	2	2			
J_5	0	0	2	4	3	3			

Find need matrix and calculate the safe sequence by using Banker's algorithm. Mention the above system is safe or not safe. (8)

13) Define the necessary and sufficient conditions for deadlock to occur. (4)

14) Consider the following snapshot of a system



Process	Allocated resources			Maximum requirements			Total resources		
	R_1	R_2	R_3	R_1	R_2	R_3	R_1	R_2	R_3
P_1	2	2	3	3	6	8	7	7	10
P_2	2	0	3	4	3	3			
P_3	1	2	4	3	4	4			

i) What is the content of Need Matrix?

ii) Is the system in safe state?

iii) If the following requests are made, can they be satisfied / granted immediately in the current sate: P1 requests for (1,1,0)

P3 requests for(0,1,0) resources additionally. (10)

15) Given three processes A,B and C , 3 resources X,Y and Z and the following events:

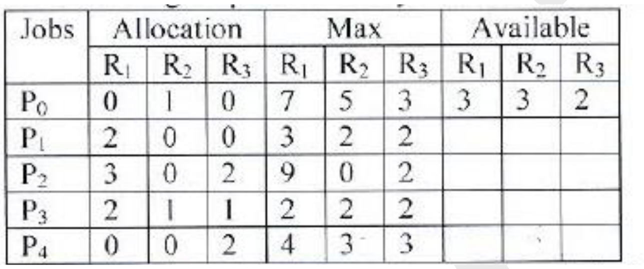
i) A requests X ii) A requests Y iii) B requests Y iv) B requests Z
v) C requests Z vi) C requests X vii) C requests Y.

Assume the requested resource is always allocated to requesting process if it is available. Draw the resource allocation graph for the sequence (2,6,3,5,1,4 and 7). Also mention if deadlock occurs. If so, how do you recover from the deadlock? (6)

<u>June 2012</u>

16) What is **deadlock**? Explain the necessary conditions for its occurrence. (6)

17) System consists of five processes(P0,P1,P2,P3,P4) and three sources (R1,R2,R3). Resource type R1 has 10 instances, resource type R2 has 5 instances and R3 has 7 instances. The following snapshot of the system has been taken:



Jobs	Allocation			Max			Available		
	R_1	R_2	R_3	R_1	R_2	R_3	R_1	R_2	R_3
P_0	0	1	0	7	5	3	3	3	2
P_1	2	0	0	3	2	2			
P_2	3	0	2	9	0	2			
P_3	2	1	1	2	2	2			
P_4	0	0	2	4	3	3			

Calculate the content of matrix need and find out safe sequence by using Banker's algorithm.

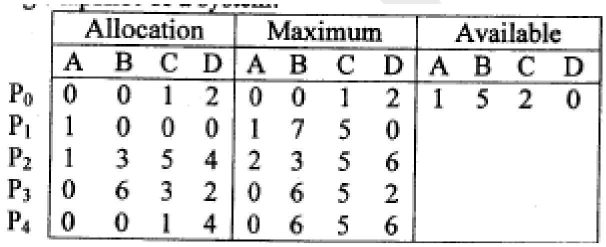
<u>Dec-2010</u>

18) Define the terms: safe state and safe sequence. Give an algorithm to find whether or not a system is in a safe state. (10)

<u>Dec-Jan 2010</u>

19) Explain Banker's algorithm for deadlock avoidance. (6)

20) Consider the following snapshot of a system:



	Allocation				Maximum				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	0	0	1	2	0	0	1	2	1	5	2	0
P_1	1	0	0	0	1	7	5	0				
P_2	1	3	5	4	2	3	5	6				
P_3	0	6	3	2	0	6	5	2				
P_4	0	0	1	4	0	6	5	6				

Answer the following questions using the Banker's algorithm:

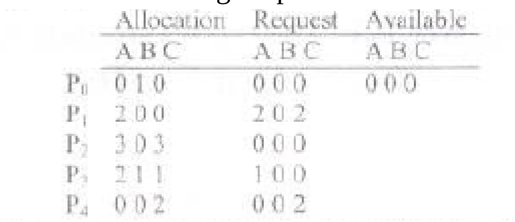
i) What is the content of matrix need?

ii) Is the system in a safe state?

iii) If a request from process P1 arrives for(0,4,2,0) can the request be granted immediately?(10)

Dec -Jan09

21) Consider the following snap shot of resource-allocation at time t1



	Allocation	Request	Available
	A B C	A B C	A B C
P_0	0 1 0	0 0 0	0 0 0
P_1	2 0 0	2 0 2	
P_2	3 0 3	0 0 0	
P_3	2 1 1	1 0 0	
P_4	0 0 2	0 0 2	

i) Show that the system is not deadlocked by generating one safe sequence.

ii) At instance t2, P2 makes one additional request for instance of type C. Show that the system is deadlocked if the request is granted. Write down the deadlocked processes. (10)

Dec-Jan08

22) Why is deadlock state more critical than starvation in a multiprogramming environment. Describe a resource allocation graph, i) with a dead lock ii) with a cycle but no deadlock. (8)

23) What are the two options for breaking a deadlock? Explain each clearly. (7)

24) What is wait-for graph? Explain how it is useful for detection of deadlock. (5)

25) What are the different ways to prevent the occurrence of deadlock

26) Explain the Banker's algorithm

Memory Management

June/July 2017

1. Explain the multistep processing of a user program with a neat block diagram. (5)

2. Distinguish between internal and external fragmentation. (2)

3. Explain segmentation with an example. (6)

4. Consider the following segment table: (7)

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

What are the physical address for the following logical address?

i) 0,430 ii) 1,10 iii) 2,500 iv) 3,400 v) 4,112

May 2017

1 Explain internal and external fragmentation with examples. (6)

2 Explain with a diagram, how TLB is used to solve the problem of simple paging scheme. (8)

3 What is thrashing? How does the system detect thrashing? (6)

Dec-2015

1) Discuss paging with an example. (8)

2) Consider the following page reference string
1,2,3,5,2,3,5,7,2,1,2,3,8,6,4,3,2,2,3,6.
Assuming there are three memory frames, how many faults would occur in the case of
i) LRU ii) Optimal algorithm.
Note that initially all frames are empty. (6)
3) What is thrashing? Explain. (6)

June-2015

4) With a supporting paging hardware, explain in detail concept of paging with an example for a 32-byte memory with 4-type pages with a process being 16-bytes. How many bits are reserved for page number and page offset in the logical address. Suppose the logical address is 5, calculate the corresponding physical address, after populating memory and page table. (10)
5) Discuss on the performance of demand paging. (5)
6) What is Belady's anomaly? Explain with an example. (5)

Dec - 2014

7) What is locality of reference ? (2)

Dec –Jan2013

8) Why are **translation look-aside buffers**(TLB) important? In a simple paging system what information is stored in TLB? Explain. (8)
9) Given memory partitions of 100K,500K,200K,300K and 600K, apply first fit and best fit algorithm to place 212K,417 K,112 K and 426 K. (4)
10) What is **swapping**? Does this increase the operating systems overhead? Justify your answer. (8).

June-July 2013

11) What are the draw backs of contiguous memory allocation?
12) Consider a paging system with the page table stored in memory.
i) if a memory reference takes 200 nano seconds, how long does a paged memory reference take?
ii) if we add associative register and 75 percentage of all page table references are found in the associative registers, what is the effective memory access time? (Assume that finding a page table entry in the associative memory/registers takes zero time, if the entry is found).

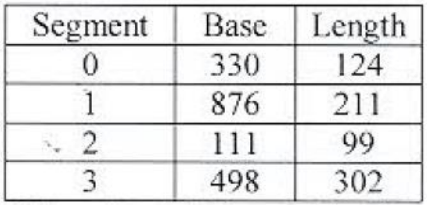
June 2012

13) Distinguish between:
i) Logical address space and physical address space.
ii) Internal fragmentation and external fragmentation.
iii) Paging and segmentation. (6)
14) Explain with the help of supporting hardware diagram how the TLB **improves the performance** of a demand paging system. (10)

15) Given memory partitions of 100K,500K,200K,300K and 600K (in order) how would each of the first fit, best fit and worst fit algorithms work place processes of 212K,417 K,112 K and 426 K (in order)? Which algorithm makes the most efficient use of memory? (4)

Dec 2012

16) On a system using simple segmentation, compute the physical address for each of the logical address, logical address is given in the following segment table. If the address generates a segment fault, indicate it as “segment fault”



Segment	Base	Length
0	330	124
1	876	211
2	111	99
3	498	302

i)0,9,9 ii)2,78 iii)1,265 iv) 3,222 v) 0,111 (5)

17) Short notes on:
a. Page replacement algorithms
b. Steps in handling page fault.

18) Explain with illustration, the internal and external fragmentation problem encountered in contiguous memory allocation. (6)

19) Explain the concept of forward mapped page table. (6)

20) Consider the following sequence of memory references from a 460 word program.
10,11,104,170,73,309,185,245,246,434,458,364
i) Show the reference string assuming page size of 100 words
ii) Find page fault rate for the above reference string assuming 200 words of primary memory available and FIFO and LRU replacement algorithms. (8)

June-July 2011

21) Consider the following page reference string.
1, 2, 3, 5, 2, 3, 5, 7, 2, 1, 2, 3, 8, 6, 4, 3, 2, 2, 3, 6.
How many page faults would cocur in the case of
i) LRU
ii) FIFO
iii)Optimal algorithms assuming 3 frames.
(Intially all frames are empty). (8)

Dec-2011

22) What is dynamic storage allocation? Explain the commonly used strategies for dynamic storage allocation. (12)

23) Explain the buddy- system, used for managing free memory assigned to kernel process.(8)

Dec -2010

24) What do you mean by address binding? Explain with the necessary steps, the binding of instructions and data to memory addresses. (8)
25) On a system using demand paged memory it takes 0.12us to satisfy a memory request, if the page is in memory. If the page is not in memory the request takes 5000us. What would the page fault rate need to be to achieve an effective access time 1000us? Assume the system is only running a single process and the CPU is idle during the page swaps. (8)
26) What do you mean by a copy-on-write? Where is it used? Explain in brief. (4)

File System, Implementation of File System

June/ July 2017

1. Explain briefly the various operations performed on files. (6)
2. Explain the various access method of files. (6)
3. Explain various allocation methods in implementing file systems. (8)

May 2017

1. What is a file? Explain the different allocation methods. (10)
2. Explain different approaches to managing free space on disk storage. (10)

Dec-2015

1) Explain different file access methods. (6)
2) Describe the various directory structures. (8)
3) Write a note on any four different methods for managing free space. (6)

June-2015

4) With supporting diagrams distinguish between single-level and two-level directory structure.(5)
5) Compare contiguous and linked allocation methods for disk space. (5)
6) Explain bit vector free-space management technique. (5)

Dec –Jan2014

7) What is a **file**? Explain the different **allocation** methods. (10)
8) What are **directories**? Write a brief note on **mounting** file systems. (5)
9) How is **free space managed**? Explain (5)

June-July-2013

10) Explain the various file operations supported by the operating system, also differentiate mandatory lick and advisory lock mechanisms used on files by the operating system. (5)
11) Explain various file protection mechanisms. (7)

June 2012

12) Explain the different **types** of files. (5)
13) Name the different **file allocation** methods. Explain the linked allocation of file implementation with merits and demerits. (8)
14) Define **file system**. Explain the different directory structures. (7)

<u>Dec 2012</u>

15) Explain the different types of **directory structures**, with examples and mention their advantages and disadvantages. (8)
16) With supporting diagrams, explain **linked and indexed** method of allocating disk space (8)
17) Describe the different **access method**s on files. (9)

<u>Dec-2011</u>

18) How do the modern operating systems concurrently support multiple types of file system? Explain its implementation in detail. (10)

<u>Dec-Jan 2010</u>

19) Explain the following:

i) File Types.
ii) File operations
iii) File attributs.
iv) Tree directory structure.
v) Thrashing
vi) Monitors.
vii) Acyclic-graph direcotory.

<u>Dec -2010</u>

20) What do you mean by a **free space list**? With suitable examples, explain any two methods of implementation of a free space list.

<u>May-June 2010</u>

21) What is meant by 'consistency semantics'. Explain the consistency semantics as implemented in a modern OS (7)
22) Explain Virtual File System.(VFS) (5)

Secondary Storage Structures, Protection

<u>June/July 2017</u>

1. Explain the various Disk scheduling algorithms with example. (10)
2. Point out and explain briefly the problems with RAID (5)
3. Explain Access Matrix method of system protection. (5)

May 2017

1. What is disk scheduling? Explain the following with diagram: i) FCFS ii) SSTF iii) SCAN. (10)
2. What is an access matrix? Explain the following operations in access matrix with an example for each:
   i) Copy ii) Transfer iii) Limited copy (10)

Dec-2015

1) Suppose the position of cylinder is at 53. The disk drive has cylinders numbered from 0-199. The queue of pending requests in FIFO order is: 98,183,37,122,14,124,65,67. Starting from the current head position, what is the total distance traveled (in cylinders) by the disk arm to satisfy the requests using algorithms FCFS, SSTF, SCAN and LOOK. Illustrate with figures in each case. (12)
2) Describe the access matrix model used for protection purpose. (8)

June-2015

3) With an example, distinguish between SSTF, FCFS,SCAN and LOOK DISK scheduling. (8)
4) What are boot block and bad blocks? Explain. (6)
5) Explain the goals and principles of protection. (6)

Dec –Jan2014

1) Explain the difference between protection and security? Describe the scheme of capability lists to implement protection (10)
2) Write short notes on:
   i) **Swap space** management
   ii) **Revocation** of access rights. (10)

June 2012

3) What is **disk scheduling**? Explain any 3 disk scheduling methods with examples.(10)
4) What is **access matrix**? Explain access matrix with domains as objects. (5)

Dec 2012

5) Briefly explain the **selection** of a disk scheduling algorithm (4)
6) What is **protection**? Distinguish between mechanisms and policies. Explain briefly the access matrix with domains as objects. (7)
7) A drive has 5000 cylinders numbered 0 to 4999. The drive is currently serving a request at 143 and previously serviced a request at 125. The queue of pending requests in FIFO order is : 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130 starting from current head position, what is total distance travelled(in cylinders) by disk arm to satisfy the requests using FCFS,SSTF,SCAN and look algorithms.
   (10)

Dec-Jan 2010

8) Explain SCAN and C-SCAN disk scheduling. (6)

Case Study: The Linux Operating System

<u>June/July 2017</u>

1. Explain the various components of a Linux system. (6)

2. Explain process scheduling in a Linux system. (6)

3. Explain file systems implementation in Linux. (8)

<u>May 2017</u>

1. Explain the different components of a Linux system. (10)

2. Discuss how memory management is dealt with in Linux operating system. (10)

<u>Dec-2015</u>

1) Write short note on: a) process management in Linux. b) Linux filesystem. c) Benefits of multi-threading. d) Inter-process communication. (20)

<u>June-2015</u>

1) Write a short note on Linux virtual memory system.

<u>Dec –Jan2014</u>

1) What are the **design** principles of Linux operating systems? Explain (8)

2) What do you mean by **cloning**? How is it achieved in Linux systems? (6)

3) How is **IPC** handled in Linux? Explain with a suitable example (6)

<u>June 2012</u>

4) Write short notes on:

i) Process management in Linux. (6)

<u>Dec 2012</u>

5) Write short notes on :

a. Linux history

b. Components of a Linux system

<u>Dec-2011</u>

6) Explain the Linux device drive block structure. (12)