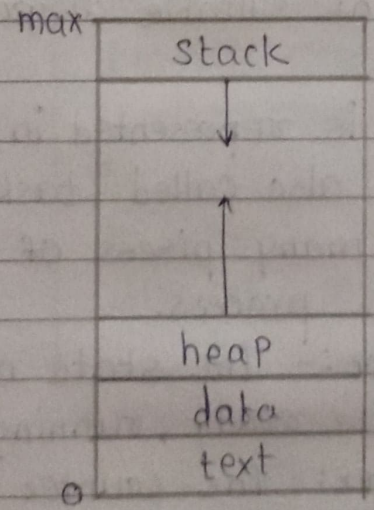
Page No. YOUVA
Date

Assignment No. 2

Q.1 What is process? explain the process concept in detail.

→ (1) A process can be thought of as a program in execution.
(2) A process is unit of work in most systems.
(3) Process concepts :-
(i) A process is more than program code, which is sometimes known as text section.
(ii) It also includes the current activity, as represented by value of the program counter & the contents of processor's register.
(iii) A process generally includes the process stack, which contains temporary data.
(iv) A process may also include a heap, which is memory that is dynamically allocated during process run time.



process in memory

(v) A program by itself is not a process.
(vi) A program is passive entity, such as a file containing a list of instruction stored on disk.
(vii) a process is an active entity, with program

counter specifying the next inst. to execute of a set of associated resources.

(viii) A program becomes a process when an executable file is loaded into memory.

(ix) Two common techniques for loading executable files:-

(A) double-clicking an icon representing the executable file.

(B) entering the name of executable file on command line

(x) Although two processes may be associated with same program, they are nevertheless considered two seperate execution sequences.

(xi) For instance, several users may be running different copies of mail program.

(xii) The same user may invoke many copies of web browser program.

Q. 2] With the help of suitable diagram explain PCB.

→ (1) Each process is represented in OS by process control block also called task control block.

(2) It contains many pieces of info. associated with specific process.

(3) Process state:- The state may be new, ready, running, waiting, halted so on.

(4) Program counter:- The counter indicates the address of next inst. to be executed for this process.

(5) CPU register:- The registers vary in number & type, depending on computer architecture. They include accumulator,

index registers, stack pointers & general purpose register, plus any condition-code information.

(6) CPU-scheduling information: This information includes a process priority.

(7) Memory-managment information:- This information may include such items as the value of the base & limit registers & page tables or the segment tables, depending on memory system used by OS.

(8) Accounting information :- This information includes the amount of CPU & real time used, time limits, account numbers, job or process number & so on.

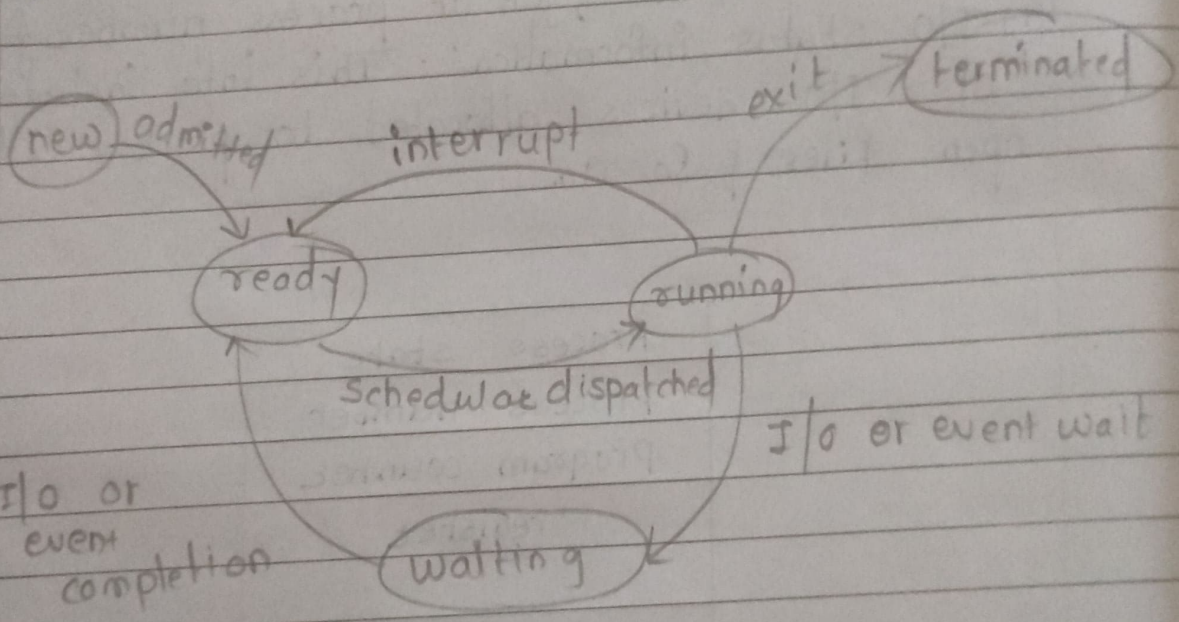
(9) I/O status information :- This info. includes the list of I/O devices allocated to process a list of open files & so on.

Process state
Process number
Program counter
registers
Memory limits
list of open files
...

Process control block (PCB)

Q. 3] Explain process states using process state diagram.

→ ① As a process executes, it changes state.
A] New :- The process is being created.
B] Running :- Instruction are being executed.
C] Waiting :- The process is waiting for some event to occur
D] Ready :- The process is waiting to be assigned to a processor.
E] Terminated :- The process has finished executio



Q. 4] What is scheduler? explain the different type of scheduler in OS.

→ ① A process migrates among the various scheduling queues throughout its lifetime.
② Schedulers are special system software which handles process scheduling in various ways. Their main task is to select job

to be submitted into the system & to decide which process to run.

(3) There are three types of schedular :-

A] Long Term schedular

B] Short Term schedular

C] Medium Term schedular

A] Long Term ~~schedular~~ schedular :-

(i) long Term schedular runs less frequently.

(ii) It is also called job schedular. A long-Term schedular determines which programs are admitted to system for processing.

(iii) Long-Term schedular decide which program must get into job queue.

(iv) From job-queue, the job processor, select processes & loads them into the memory for execution.

(v) Primary aim of job schedular is to maintane good degree of multiprogramming.

(vi) An optical degree of multiprogramming means the average rate of process creation is equal to average departure rate of processes from execution memory.

B] short-term schedular :-

(i) short-term scheduling involves selecting one of the processes from the ready queue & scheduling them for execution. This is done by short-term schedular. A scheduling algorithm is used to decide which process will be scheduled for execution next by short-term scheduler.

(ii) The short-term schedular executes much more frequently than long-term schedular as a process may execute only for a few millisecond.

(iii) The choices of short term schedular are very important. If it select a process with long burst time, then all the processes after that will have to wait for long time in ready queue. This is known as starvation & it may happen if a wrong decision is made by short-term schedular.

c] Medium Term Schedular :-

(i) Medium-Term scheduling involves swapping out a process from main memory. The process can be swapped in later from the point it stopped executing. This can also be called as suspending & resuming the process & is done by medium-term schedular.

(ii) At some later time, the process can be reintroduced into memory & its execution can be continued where it left off.

(iii) This scheme is called swapping. The process is swapped out, & is later swapped in, by medium term schedular.

Q. 5] Explain various types of queues with the help of queueing diagram.

→ (1) Job queue :- As processes enter the system, they are put into a job queue, which consists of all processes in the system.

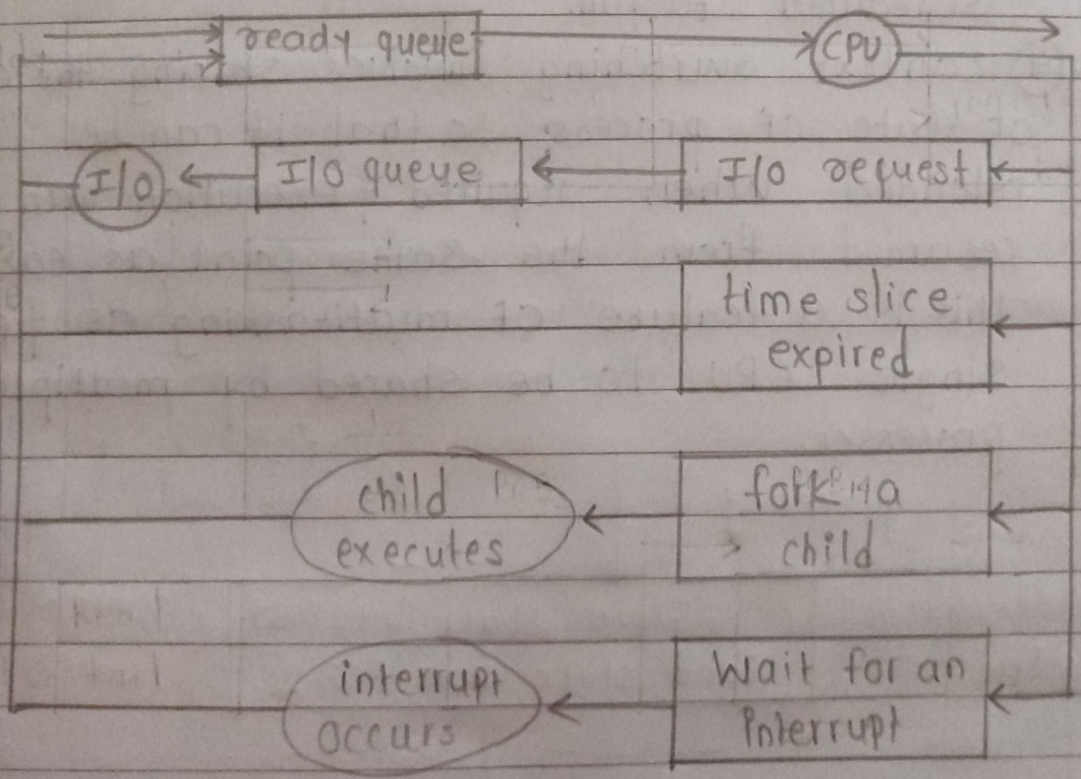
(2) Ready queue: (i) The processes that are residing in main memory & are ready & waiting to execute are kept on list called ready queue.

(ii) This queue is generally stored as linked list. A ready-queue header contains pointers to the first & final PCB in the list.

(iii) Each PCB includes a pointer field that points to the next PCB in ready queue.

(3) Device queue: - (i) The system also includes other queues. The list of processes waiting for a particular I/O device is called device queue.

(ii) each device has its own device queue.



Q. 6] Write a note on:

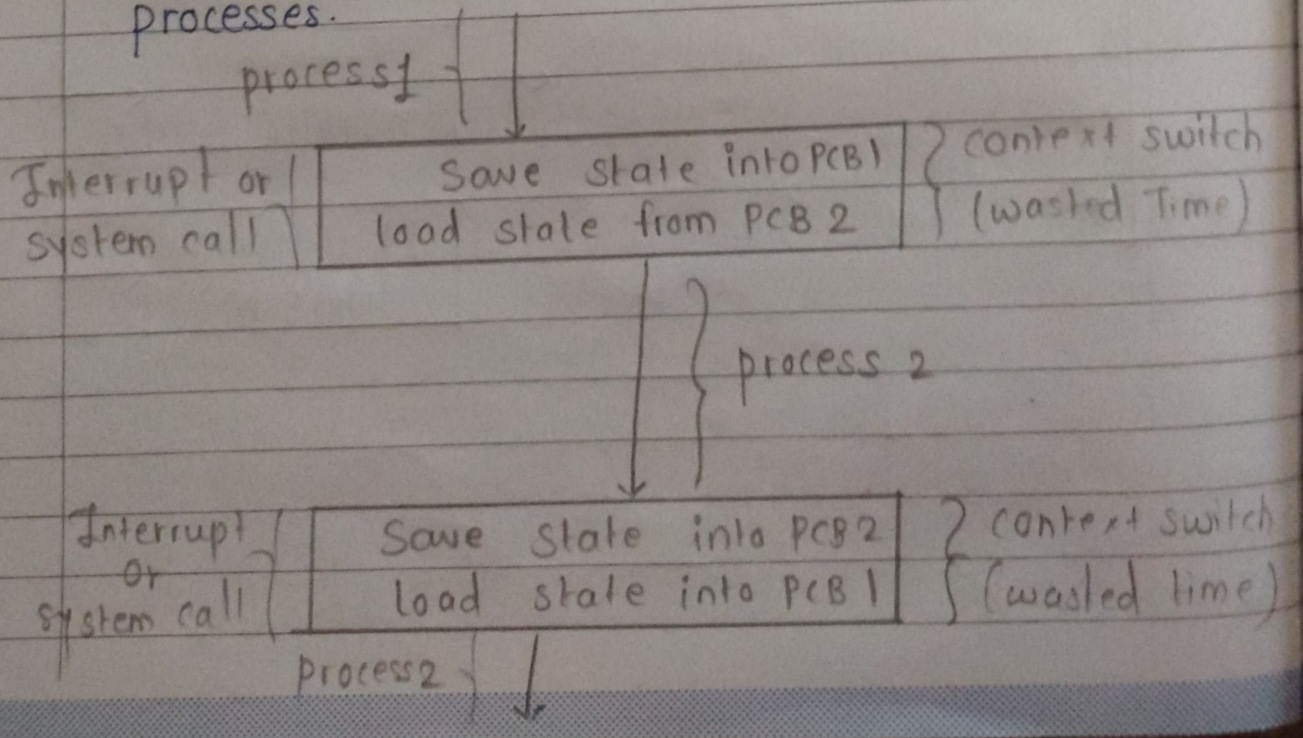
a] context switch :-

① Switching the CPU to another process requires saving the state of old process & loading the saved state for new process. This task is known as context switch.

② The context of a process is represented in the Process control Block of process; it includes the value of CPU registers, the process state & memory managment information.

③ When context switch occurs, the kernal saves the context of old process in its PCB & loads the saved context of the new process scheduled to run.

④ context switching involves storing the context or state of process so that it can be reloaded when required & execution can be resumed from the same point as earlier. This is a feature of multitasking os & allows single CPU to be shared by multiple processes.



b] Shared memory model :-

① Inter-process communication using shared memory requires communicating processes to establish a region of shared memory

② Typically, a shared-memory region in the address space of the process creating the shared-memory segment.

③ Other processes that wish to communicate using the shared-memory segment must attach it to their address space.

c] Message passing model :-

① Message passing provides a mechanism to allow processes to communicates & to synchronize their actions without sharing the same address space.

② It is particularly useful in a distributed environment, where the communicating process may reside on different computers connected by a network.

③ An Internet chat program could be designed so that chat participants communicate with one another by exchanging messages.

d] RPC :-

① The RPC was designed as a way to abstract the procedure-call mechanism for use bet' systems with network connections.

② It is similar in many respects to the IPC mechanism however, because we are dealing with an environment in which the processes are executing on seperate systems, we must use a message-based communication scheme to provide remote service.

③ In contrast to IPC messages, the messages exchanged in RPC communication are well structured & are thus no longer just packets of data.

④ each message is addressed to an RPC daemon listening to a port on the remote system, & each contains an identifier specifying the fun^n. to execute & the parameters to pass to that fun^n.

⑤ The function is then executed as requested & any output issent back to the requester in separate message.

e] Scheduling criteria :-

① Different CPU-scheduling algorithm have different properties.

② Many criteria have been suggested for comparing CPU-scheduling algorithm.

③ criteria includes:-

1] Throughput: If The CPU is busy executing processes, then work is being done.

one measure of work is the number of processes that are completed per time unit, called throughput. For long processes, this rate may be one process per hour for short transactions, it may be ten processes per second

2] Turnaround Time :- The interval from the time of submission of a process to the time of completion is turnaround time.
Turnaround time = period spent waiting to get into memory + waiting in the ready queue + executing on CPU + and doing I/O

3] Waiting time :- Waiting time is the sum of periods spent waiting in the ready queue.

4] The CPU-scheduling alg. affects only the amount of time that a process spend waiting in ready queue.

4] Response Time :- The time from the submission of request until the first response is produced.

f] Real-time CPU scheduling :-
① CPU scheduling for real time operating system involves special issues.
② We can distinguish bet^{n}. soft real-time system & Hard real-time systems.
③ i) Soft real-time systems provide no guarantee as to when a critical real-time process will be scheduled.
ii] They guarantee only that the process will be given preference over noncritical processes.

④ Hard real-time systems : It have stricter requirments. A task must be serviced by its deadline ; service after the deadline has expired is the same as no service at all.

Q. 7] explain any two CPU-scheduling algorithms with suitable examples.

→ ① First-come, First-served scheduling :

A] The simplest CPU-scheduling algo. is the first-come, first-served scheduling alg.

B] With this scheme, the process that request the CPU first is allocated the CPU first.

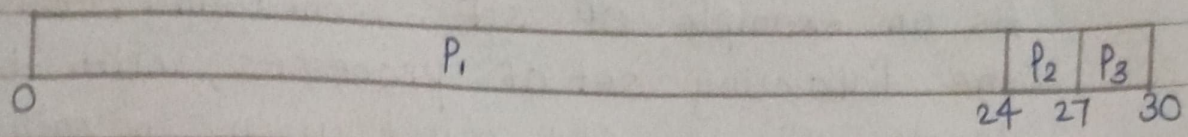
C] The implementation of FCFS policy is easily managed with FIFO queue.

D] What a process enters the ready queue, its PCB is linked into the tail of the queue. When the CPU is free, it is allocated to the process at the head of the queue.

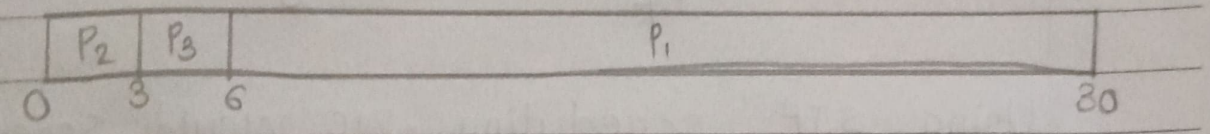
e] The running process is then removed from the queue.

f] Consider the following set of processes that arrive at time 0, with the length of the CPU burst given in milliseconds :

Process	Burst time
P_1	24
P_2	3
P_3	3



The waiting time is 0 miliseconds for process P_1, 24 miliseconds for process P_2 & 27 miliseconds for process P_3. Thus, the average waiting time is $(0+24+27)/3 = 17$ miliseconds. If the processes arrive in order P_2, P_3, P_1, however, the results will be as shown in following Grantt chart:



The average waiting time is now $(6+0+3)/3 = 3$ miliseconds. This reduction is substantial. Thus, the average waiting time under an FCFS policy is generally not minimal & may vary substantially if the processes CPU burst times vary greatly.

(2) Shortest-Job-First Scheduling:

(1) shortest Job First scheduling works on the process with the shortest burst time / duration first.

(2) When the CPU is available, it is assigned to the process that has the smallest next CPU burst.

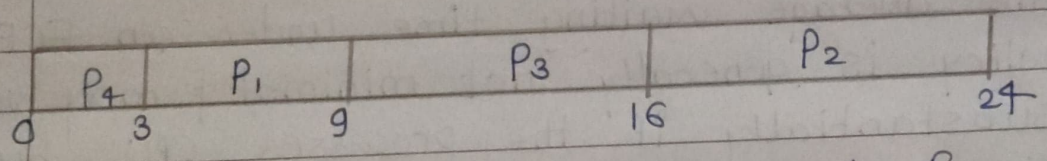
(3) If the next CPU burst of two processes are the same, FCFS scheduling is used to break the tie.

(4) More appropriate term for this scheduling method would be the shortest-next-CPU-burst alg., because scheduling depends on length of the next CPU burst of process.

As an example of SJF scheduling, consider the following set of processes, with the length of the CPU burst given in miliseconds:

Process	Burst time
P_1	6
P_2	8
P_3	7
P_4	3

Using SJF scheduling, we would schedule these processes according to the following Grantt chart:



The waiting time is 3 miliseconds for process, P_1, 16 miliseconds for process P_2, 9 miliseconds for process P_3 and 0 miliseconds for process P_4. Thus, the average waiting time is $(3+16+9+0)/4 = 7$ milliseconds. By comparison, if we were using the FCFS scheduling scheme, the average waiting time would be 10.25 miliseconds.

(5) To successfuly implement it, the burst time/duration time of processes should be known to processor in advance, which is practically not feasible all the time.

(6) This scheduling alg. is optimal if all the processes are available at the same time.