Meet A. Raut
S21 84

Assignment 1

Q.1) Given a system with n processes, how many possible ways processes can be scheduled?

Ans. The number of possible ways n processes can be scheduled depends on the scheduling algo used. For eg. in preemption scheduling algo., each process can be interrupted and switched out leading to different interlearning. The number of permutations can be calculated as n factorial where $n! = n(n-1)(n-2)\ldots \times 2 \times 1$ where n is a processes

1. FCFS :

In this method the process are executed in the order they arrive. Only one scheduling time can be determined. Here it may lead to starvation.

2. SJF :

Here the process having smallest burst time or execution time are executed first. In non premptive SJF, a ~~person~~ process with shorter burst time or execution time. This method of scheduling usually maximizes waiting & turnaround time.

3. RR :

The processes are assigned a fixed time quantum for execution. Processes are excited in a cyclic manner & the scheduling process depending on factors such as the time quantum & process order.

4. Priority scheduling :

Processes are executed based on their priority. If the higher priority process are executed first. However this may lead to starvation for lower priority processes.

Q.2) Define OS. Enlist functions of OS.

Ans. An OS acts as an inder~~[illegible]~~face between the user and the hardware of the computer and also controls the execution of the application programs. OS is also called as resource manager. The most important functions of the OS are:

i) Process Management:

Createon, execution and deletion of user and system processor syncronization, inter process communication and deadlock handling for processes.

ii) Memory management:

Allocating primary as well as secondary memory to user and ~~processe~~ ~~[illegible]~~ system processes

iii) File management:

Creation and deletion of files and directories and backup of file.

iv) Device management:

Keep an eye on device driver communicate, control and monitor the device drive

v) Protection and security:

Provide user authentication, files attributes such as read, write, encryption and back-up.

Q. State 2 advantages of kernel level threads among the two thread types which one is better?

Ans.

Two advantages of kernel-level threads are:

(i) Concurrency handlings : kernel level threads can run in parallel on multiple processors or core, allowing for efficient utilization of system resources and improved overall system performance.

ii) Independance from user-level code:
kernel level threads are managed by the OS kernel making them less dependent on user level code. This leads ~~th~~ to better stability and relaibability because the kernel can manage thread execution and resources more effectively

kernel level threads after advantages in terms of concurrency and system control by may have higher overhead. User level thread provide more flexibility are often eighter in terms of ~~more~~ resource usage but might not take full advantage of multi core systems.

Q. Define PCB. Enlist 3 components of PCB.

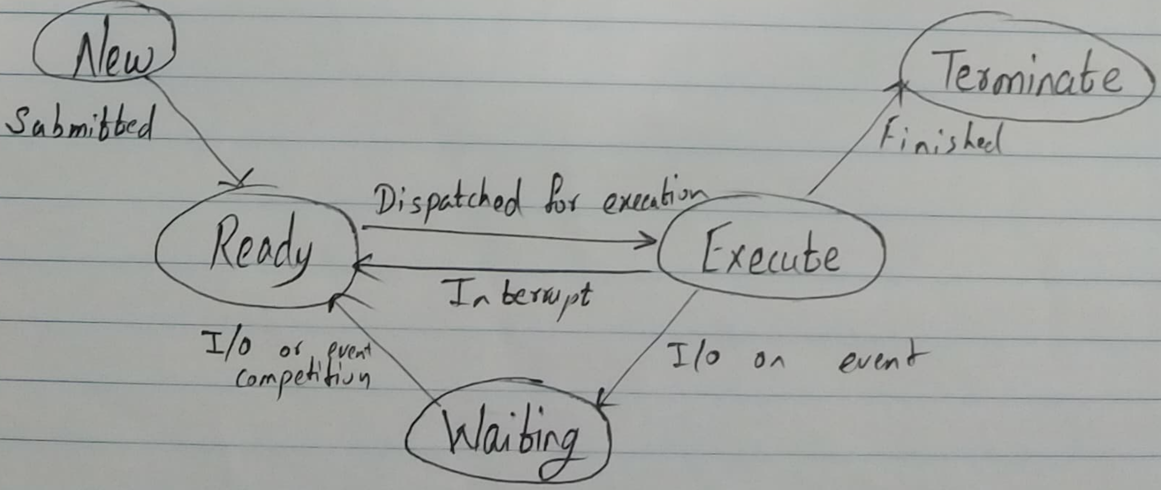
Ans. Any process is identified by its process control block (PCB). PCB is data structure used for keeping track on the process by OS. All the ~~in~~ informetic associated with the process is kept in its PCB. There is a seperate PCB for each process.
The three components of PCB are:

i) Pointer:
This field points to other processes PCB. The scheduling list is maintained by pointer.

ii) Current State:
Currently process can be in any of the state from new ready, executing, waiting, etc.

iii) Process IW:
Identification number of process. OS assign this number to process to distinguish it from ~~p~~other processes.

Q. Sketch 5 state process state transition model.



Q For each of the following indicate wether transition is possible or not if yes, explain.

Ans.

- Executing/Running → Ready

yes, this transition is possible, whenever an interrupt is generated the processes goes back in the . Ready Queue and stops executing.

- Executing/ ~~Running~~ waiting →

yes, whenever the process in running and require an I/o or a event to occur to continue the process than it enters waiting state to wait for I/0 or event to occur.

- Waiting → Execute

No, when as explained above the process after event/ I/o competition will transition to ready state first and then to execution state.

- Terminated → waiting.

No, after termination it cant change to any state.

Q. State one principle of concurrency in detail.

Ans.

The ~~process~~ ~~of~~ principle of process synchronization :

In a single processor multiprogramming system processors are not executed concurrently. In order to get the appearance of concurrent execution, a fixed time slot is allocated to each, process. After utilization of each time slot, CPU gets allocated to another process. Such switching of CPU back and forth between processes is called as context switch. At a time single process gets executed so parallel processing cannot be accomplished. Also there is definite amount of overhead drawn in switching back and forth between processes. Apart from above limitation, interleaved execution ~~often~~ offers major benifit in processing efficiency and in program-structuring in a multi processing system, intrleaving and overlapping of multiple processes is achievable The comparative speed of execution of processes depends on activities and behaviour of other processes.

Q. For the processes listed, draw gantt chart of these processes using FCFS, SJF, find avg waiting and avg turnaround time.

Ans.

For FCFS :

Process	Arrival Time	Burst Time	Completion Time	TAT	WT
A	0.001	3	3	3	0
B	1.001	8	11	10	2
C	4.001	4	15	11	7
D	6.001	2	17	11	9

Gantt chart :

A	B	C	D

0 3 11 15 17

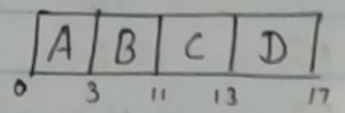
Avg turnaround time = (3 + 10 + 11 + 11) / 4
= 8.75

Avg. waiting time = (0 + 2 + 7 + 9) / 4
= 4.5

ii) SJF

Process	AT	BT	CT	TAT	WT
A	0.001	3	3	3	0
B	1.001	8	11	10	2
C	4.001	4	17	13	9
D	6.001	2	13	7	5

gantt chart :



Avg. TAT = (3 + 10 + 13 + 7) /4
= 5.75

~~Avg.~~

Avg. WT = (0 + 2 + 9 + 5)/4
= 4