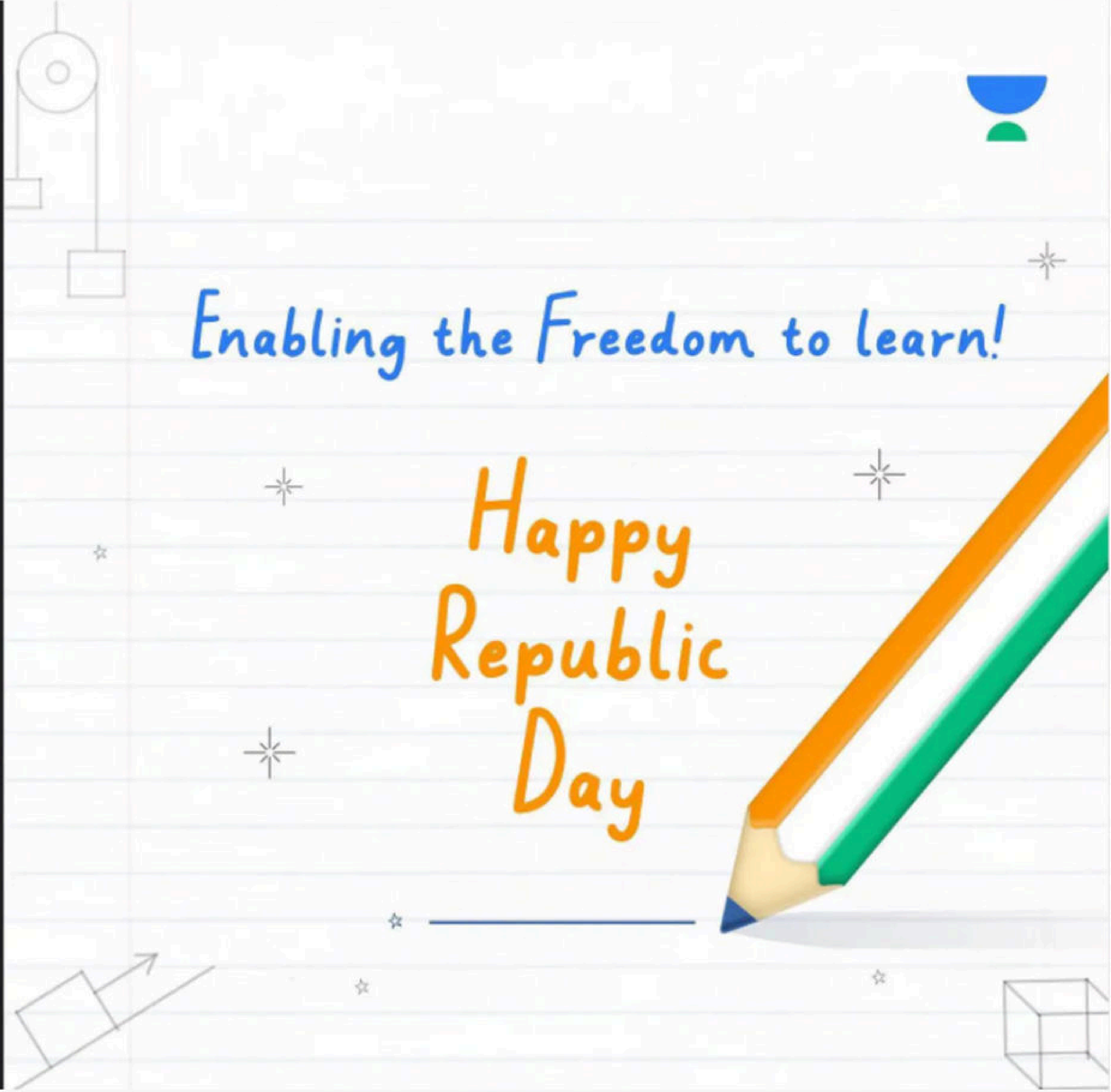
Short Notes & Formula Revision: OS

Special class

Vishvadeep Gothi • Jan 26, 2022



Enabling the Freedom to learn!

Happy
Republic
Day

~~D~~OS: Short Notes

By: Vishvadeep Gothi

Basics & Process

- Apart from all OS by MS & Apple, other all OS are multiuser OS
- Multitasking OS uses RR algo
- Dual Mode of Operation used to provide system protection
- A process is stored in 4 parts in main memory:
  1. Data section (Fixed Size): Global & static variables
  2. Code section(Fixed Size): Instructions
  3. Heap (Variable Size): Dynamic memory allocation
  4. Stack (Variable Size): local variables, function parameters, return addresses → activat^n record
- Attributes of process maintained and controlled by OS through PCB
- Process can not access its own PCB
- The content of PCB of a process are collectively know as ‘Context’ of that process
- A process is in main memory if it in ready, running or blocked state.
- A process is deallocated from memory when terminated or swapped out
- 2 Transitions are voluntary: Running to Terminated, Running to Blocked

Vishvadeep Gothi

Process

MTS can increase & decrease the degree of multiprog.

LTS can only increase Degree of multiprogramming.

- LTS maintains a good mixture of CPU bound & IO bound processes for better resource utilization
- STS only schedules the processes, context switch is performed by dispatcher
- Goal of scheduling:
    1. Minimize Wait time and Turn-around time
    2. Maximize CPU utilization (Throughput)
    3. Fairness (no starvation)
- FCFS suffers from convoy effect
- SJF gives minimum avg WT for non-preemptive scheduling algorithms always
- SRTF gives minimum avg WT for all scheduling algorithms always
- RR is used for interactiveness
- SJF, SRTF, Priority based algo suffer from starvation (LRTF also)
- SRTF behaves same as SJF if all processes arrive together.
- Preemptive and non- Preemptive priority based algo behave same, if all if all processes arrive together.
- Avg WT SJF ≤ Avg WT FCFS

Process

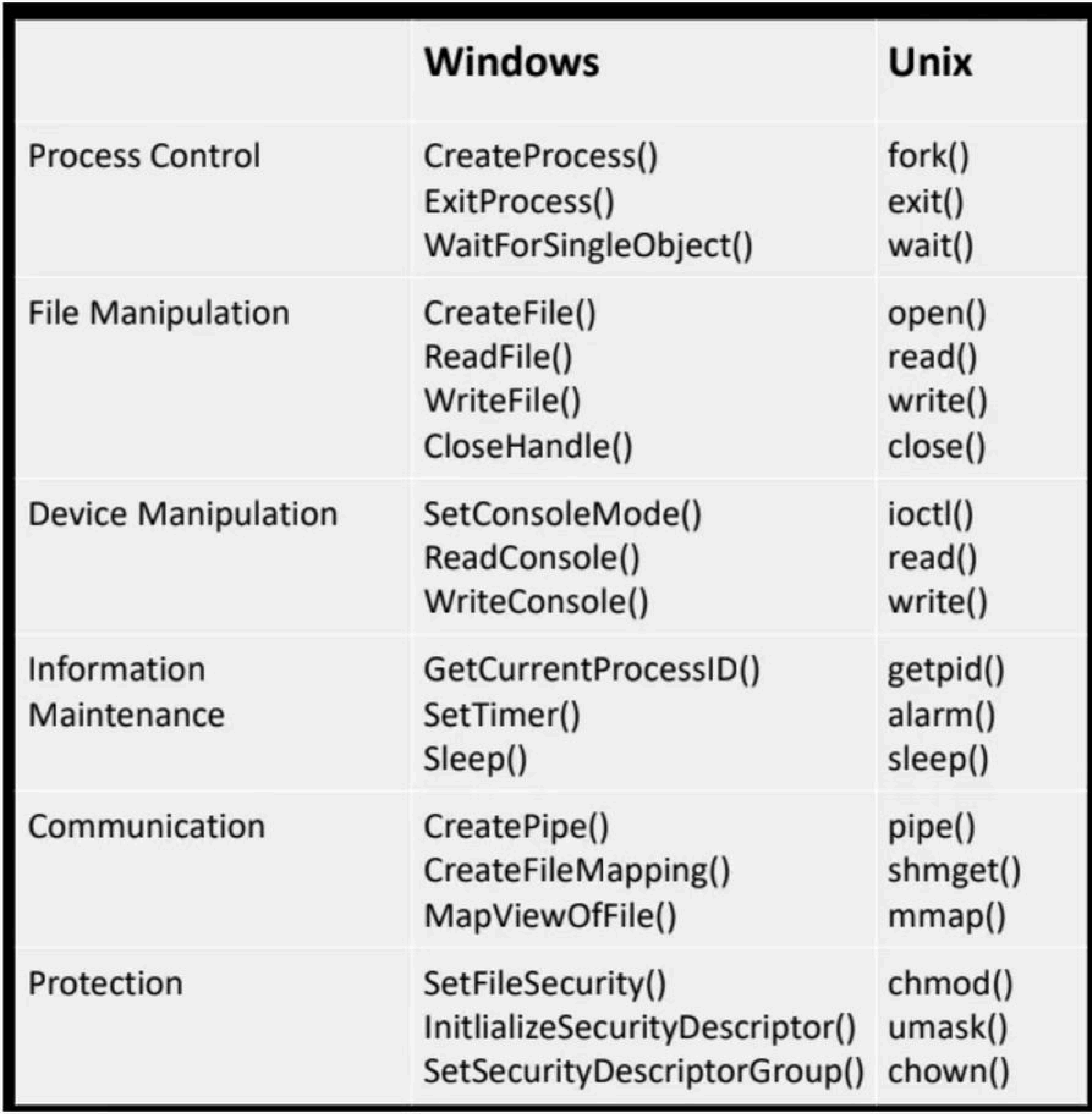
- In RR if Q → very small ⇒ efficiency = 0 (so many context switches)
  → very large ⇒ RR degrades to FCFS.
- RR provides minimum average response time
- CPU utilization = 1- P^n, where P is probability of IO, and n is number of processes

Shared Among Threads	Unique For Each Thread
Code Section	Thread Id
Data Section	Register Set
OS Resources	Stack
Open Files & Signals	Program Counter

User Threads	Kernel Thread
Multithreading in user process	Multithreading in kernel process
Created without kernel intervention	Kernel itself is multithreaded
Context switch is very fast	Context switch is slow
If one thread is blocked, OS blocks entire process	Individual thread can be blocked
Generic and can run on any OS	Specific to OS
Faster to create and manage	Slower to create and manage

Vishvadeep Gothi

System Calls



	Windows	Unix

Process Control	CreateProcess()
	ExitProcess()
	WaitForSingleObject()
		fork()
		exit()
		wait()

File Manipulation	CreateFile()
	ReadFile()
	WriteFile()
	CloseHandle()
		open()
		read()
		write()
		close()

Device Manipulation	SetConsoleMode()
	ReadConsole()
	WriteConsole()
		ioctl()
		read()
		write()

Information Maintenance	GetCurrentProcessID()
	SetTimer()
	Sleep()
		getpid()
		alarm()
		sleep()

Communication	CreatePipe()
	CreateFileMapping()
	MapViewOfFile()
		pipe()
		shmget()
		mmap()

Protection	SetFileSecurity()
	InitlializeSecurityDescriptor()
	SetSecurityDescriptorGroup()
		chmod()
		umask()
		chown()

Semaphore

- Semaphore
  - Used to provide mutual exclusion → binary
  - Used to control access to resources → binary, counting
  - Solution using semaphore can lead to have deadlock
  - Solution using semaphore can lead to have starvation
  - Solution using semaphore can be busy waiting solutions
  - Semaphores may lead to a priority inversion
  - Semaphores are machine-independent
- signal(S) can run on a binary semaphore successfully if S=1 but does not increase value of S
- Some of the ways to avoid deadlock are as follows – (in dining philosopher)
  - There should be at most (k–1) philosophers on the table
  - A philosopher should only be allowed to pick their chopstick if both are available at the same time
  - One philosopher should pick the left chopstick first and then right chopstick next; while all others will pick the right one first then left one

Deadlock

Deadlock Recovery

1. Make Sure that deadlock never occur
    - Prevent the system from deadlock or avoid deadlock
2. Allow deadlock, detect and recover
3. Pretend that there is no any deadlock

- Deadlock prevention is stricter than avoidance
- Deadlock detection When all resources have single instance: Wait-for graph
    - Cycle in wait-for graph means deadlock
- Deadlock detection When resources have multiple instances: Banker's Algorithm
- To ensure no deadlock, min no of resources = $1 + \sum(Max_i - 1)$ for each process P_i

Memory Management

- Each process has its own page table
- Page table is stored on main memory in frames
- If a page table can’t be stored on single page, then multilevel paging is used
- If paging is used, then page table(add. translation) is required even if process is of 1 page
- Number of entries in page table = Number of pages in process
- Page table size = number of pages * 1 entry size
- 1 page table entry size = frame number + protection bits
- Logical address

p	d

- Physical address

f	d

- EMAT = 2*tmm {default}
- EMAT = tmm {if page table is in registers}
- EMAT = tTLB + tmm + (1-H)*tmm {if TLB is used}
- Direct mapped TLB logical address

Tag	TLB entry no.	d

- Set associative mapped TLB logical address

Tag	TLB set no.	d

- Optimal page size = $\sqrt{2 * process\ size\ * PT\ entry\ size}$

Memory Management

- Size of segment can vary, so along with base, keep limit information also
- Limit defines max number of words within the segment
- Paging suffers from internal fragmentation
- Segmentation suffers from external fragmentation
- Valid bit in page table helps to ensure hit or page fault
- EMAT = (1-p)2tmm + p* page fault service time
- EMAT = H*(tTLB + tmm) + (1-H) [tTLB+tmm + (1-p)tmm + p*Service time]
- Page fault is an internal interrupt and OS services it
- After Page fault service the current memory access instruction restarts
- FIFO page replacement policy suffers from Belady's anomaly
- Optimal page replacement policy provides minimum page faults
- While replacement only dirty page is written back to secondary memory
- Thrashing is high level paging activity
- Solution of thrashing:
  - Working Set Model, Page Fault Frequency

File System

- Free space management
  - Free List
  - Bitmap Method
- No searching in free list, but in bitmap we search for first zero
- Free list is faster in allocating a free block
- Free list size is variable, where as bitmap size is constant

	Contiguous	Linked	Indexed
File Access	Sequential, Random	Sequential	Sequential, Random
Fragmentations	External, Internal	Internal	Internal

- SSTF (Shortest Seek Time First) provides minimum cylinder moves

Happy Learning.!