Note:- This checklist is for understanding purpose, cross checking concepts only, Please dont consider in exam point of view.

OS Basics

1.What is OS?

2.Available OS in market today

3.Basic architecture of a computer

5.Difference between primary vs secondary memory

6.What is a cluster (parallel/distributed computing)

7.Symmetric vs Asymmetric clusters, Hot standby system

8.Factors influencing storage devices

9.How flash memory used as both primary & secondary?

10.What do you main by caching

11.What are the services of OS?

12.CLI vs GUI based environments

13.What is an RTOS?

14.Multitasking multiuser OS?

15.How multitasking acheived on uniprocessor system

16.What is Client Server and P2P computing environments

17.Beowulf cluster

18.Ubiquitous computing

19.Cloud computing

20.Grid Computing

20.Role of registers in CPU execution

21.Significance of the registers PC/EIp, PSW/EFLAGS

22.What do you understand by resource multiplexing

23.What are the 2 modes of execution available on modern CPUs

24.DMA

25.What is the significance of System Bus

Kernel

1.What is a kernel, importance in OS?

2.What are the parts/sub systems in the kernel space.

3. Kernel/System mode vs User mode

4. Kernel/System space vs User space

5. What is meant by Monolithic kernel

6. What is meant by Micro kernel

7. What is the concept of Modular kernel used in linux

8. What is the typical booting process of a system

9. What is the idea behind multi stage boot loader

10.What are the techniques to speedup boot process?

11.What is a device driver

13. What is the use of Hardware abstraction layer

14. Concept of exo kernel, nano/pico kernel

System Calls & Interrupts

1.What is an interrupt,few examples of interrupts.

2.Concept of ISR/Interrupt handler, Interrupt Vector Table/IRQ Table

3.What are maskable, non maskable interrupts

4.Whether interrupts can be disabled, if so what are the consequences.

3.What is a system call

4.Whether system calls are same as interrupts ?

5.How a library function/API differs from system call?

6.System calls are implemented in which languages, where they can be used

7.How parameters are passed to System calls

8.How a system call invocation is resolved by the kernel (System Call no.)

9.What are various types of system calls available and few usages

10.What is a TRAP instruction / What is meant by software interrupt.

11.Why interrupts are asynchronous and system calls, exceptions are synchronous

12.What are exceptions, few examples

13.What is the default behavior of an exception handler

14.What is the role of system call handler, system call table