**Q1.** What is the primary purpose of an Operating System?  
A) To compile programs  
B) To provide a user-friendly environment  
**C) To manage computer hardware and software resources**  
D) To control internet access

**Q2.** Which of the following is the correct order of operating system generations?  
A) Batch → Time-sharing → Multiprogramming → Real-time  
**B) Serial Processing → Batch → Multiprogramming → Time-sharing**  
C) Batch → Multiprocessing → Real-time → Serial Processing  
D) Time-sharing → Real-time → Serial Processing → Batch

**Q3.** Which of the following is **not** a core component of an operating system?  
A) File system  
B) Process scheduler  
**C) Compiler**  
D) Memory manager

**Q4.** Which of the following is **not** a service provided by the Operating System?  
A) Program execution  
B) Error detection  
**C) Text editing**  
D) I/O operations

**Q5.** An **interrupt** is:  
A) A software error  
**B) A hardware signal indicating an event that needs immediate attention**C) A system call  
D) A compiler error

**Q6.** Which of the following is **true** about system calls?  
A) They are executed in user mode  
**B) They allow user-level processes to request services from the kernel**  
C) They are only used for graphical interface  
D) They are faster than normal function calls

**Q7.** A **process** is:  
A) A passive entity like a program  
**B) An active entity with a program counter and execution context**  
C) A background thread  
D) Only an operating system component

**Q8.** Which of the following is **not** a valid state in the process life cycle?  
A) New  
B) Wait  
C) Terminated  
**D) Complete**

**Q9.** In the process life cycle, the transition from **Running → Ready** occurs when:  
A) A process terminates  
B) An I/O request is made  
**C) A higher-priority process arrives**  
D) Memory allocation fails

**Q10.** Multithreading improves performance by:  
A) Running multiple processes in different machines  
B) Using multiple processors to execute a single thread  
**C) Executing multiple threads within the same process concurrently**  
D) Creating multiple processes for the same task

**Q11.** In a multithreaded process, which of the following is **shared** among threads?  
A) Registers  
B) Program counter  
**C) Heap and code section**  
D) Stack

**Q12.** The part of the operating system that selects the next process to run is called:  
A) Dispatcher  
**B) Scheduler**  
C) Loader  
D) Interrupt handler

**Q13.** Which of the following is a non-preemptive scheduling algorithm?  
A) Round Robin  
**B) SJF (non-preemptive)**  
C) Priority (preemptive)  
D) SRTF

**Q14.** Turnaround time is:  
**A) Completion time – Arrival time**  
B) Burst time – Waiting time  
C) Arrival time – Waiting time  
D) Arrival time – Response time

**Q15.** In FCFS scheduling, the average waiting time is generally:  
A) Minimum  
**B) Maximum**  
C) Zero  
D) Equal for all processes

**Q16.** Throughput is measured in:  
A) Time units  
**B) Number of processes completed per unit time**  
C) Waiting time  
D) CPU utilization

**Q17.** Response time is:  
**A) Time from submission to first response**B) Time from submission to completion  
C) Time spent in I/O  
D) Time spent in memory

**Q18.** Preemptive scheduling is the strategy of temporarily suspending a process when:  
A) The process performs I/O  
**B) A higher priority process arrives**  
C) CPU overheats  
D) None of the above

**Q19.** Which of the following improves CPU utilization?  
A) Shorter jobs  
B) High I/O bound jobs  
**C) Multiprogramming**  
D) FCFS scheduling

**Q20.** The main problem of SJF scheduling is:  
**A) Starvation of long processes**  
B) High CPU usage  
C) Large turnaround time  
D) Unpredictable execution

**Q21.** In multilevel queue scheduling, queues are based on:  
A) Process size  
**B) Process type and priority**  
C) Memory access  
D) Instruction type

**Q22.** Which scheduling algorithm is best suited for time-sharing systems?  
A) FCFS  
**B) Round Robin**  
C) Priority  
D) SJ

**Q23.** Priority Scheduling may cause:  
**A) Starvation**  
B) Deadlock  
C) Paging  
D) Swapping

**Q24.** SRTF stands for:  
**A) Shortest Remaining Time First**  
B) Shortest Round Time First  
C) Scheduling Round Time Formula  
D) Start Run Terminate Finish

**Q25.** Which algorithm favors CPU-bound processes?  
**A) FCFS**  
B) Round Robin  
C) SJF  
D) Priority Scheduling

**Q26.** A Gantt chart is used to represent:  
A) Memory usage  
B) I/O status  
**C) Scheduling sequence**  
D) File system

**Q27.** Aging technique is used to:  
**A) Prevent starvation**  
B) Manage memory  
C) Handle deadlocks  
D) Allocate disk space

**Q28.** Round Robin uses a time slice known as:  
A) Burst time  
**B) Quantum**  
C) Cycle  
D) Span

**Q29.** Which of these algorithms can lead to the **convoy effect**?  
A) Round Robin  
B) Priority  
**C) FCFS**  
D) SJF

**Q30.** Which of these will show **context switching** more frequently?  
**A) Round Robin**  
B) FCFS  
C) SJF  
D) Priority

**Q31.** In Round Robin, if time quantum is too small, it leads to:  
A) Good throughput  
B) Low waiting time  
**C) More context switches**  
D) Less CPU usage

**Q32.** Linux uses \_\_ to decide the next process to run in normal mode:  
A) FIFO  
**B) Completely Fair Scheduler (CFS**)  
C) Round Robin  
D) SJF

**Q33.** In Linux, real-time scheduling class includes:  
A) SCHED\_OTHER  
**B) SCHED\_FIFO**  
C) SCHED\_BATCH  
D) SCHED\_IDLE

**Q34.** SCHED\_RR is:  
**A) Preemptive and real-time**  
B) Non-preemptive and normal  
C) Batch scheduler  
D) Linux boot loader

**Q35.** In Linux, which priority range is **real-time**?  
A) 0–20  
**B) 1–99**  
C) 100–139  
D) -20 to +19

**Q36.** The default scheduling policy for normal Linux processes is:  
**A) SCHED\_OTHER**  
B) SCHED\_FIFO  
C) SCHED\_RR  
D) SCHED\_BATCH

**Q37.** The critical section problem arises in:  
A) Single-process systems  
**B) Multi-threaded environments**  
C) Networked systems  
D) Disk management

**Q38.** Which of the following is a requirement for critical section problem?  
A) Mutual Exclusion  
B) Progress  
C) Bounded Waiting  
**D) All of the above**

**Q39.** Busy waiting occurs when:  
A) Threads sleep unnecessarily  
**B) A process constantly checks for a condition**  
C) Memory is full  
D) CPU is idle

**Q40.** A race condition occurs when:  
**A) Threads update shared data simultaneously**  
B) Scheduler misbehaves  
C) Memory leaks  
D) Deadlock occurs

**Q41.** The region of code accessing shared resource is called:  
A) Monitor  
**B) Critical Section**  
C) Kernel  
D) Thread

**Q42.** The test-and-set instruction is used to implement:  
A) Paging  
**B) Synchronization**  
C) Memory allocation  
D) Compilers

**Q43.** Which of the following is a synchronization primitive?  
A) Cache  
**B) Semaphore**  
C) PCB  
D) Shell

**Q44.** Mutex is short for:  
A) Multiple-user-text  
**B) Mutual exclusion**  
C) Multithread exception  
D) None of the above

**Q45.** In binary semaphore, value can be:  
**A) 0 or 1**  
B) 0 to 10  
C) Any integer  
D) None

**Q46.** A counting semaphore is useful when:  
A) Only one thread can enter  
**B) Multiple identical resources exist**  
C) No synchronization is needed  
D) Only hardware access is required

**Q47.** The Producer-Consumer problem is an example of:  
A) Memory management  
**B) Process synchronization**  
C) Scheduling  
D) File handling

**Q48.** What data structure is used for the buffer in producer-consumer?  
A) Array  
B) Stack  
**C) Queue**  
D) Linked list

**Q49.** A Monitor is:  
A) Hardware device  
**B) High-level synchronization construct**  
C) Compiler  
D) Timer

**Q50.** Monitors eliminate:  
A) Semaphores  
B) Deadlocks  
C) Context switches  
**D) Low-level thread operations**

**Q51.** Deadlock occurs when processes:  
**A) Wait indefinitely for resources**  
B) Share CPU  
C) Run in parallel  
D) Use FCFS

**Q52.** Which is **not** a necessary condition for deadlock?  
A) Mutual exclusion  
B) Hold and wait  
**C) Preemption**  
D) Circular wait

**Q53.** A cycle in a resource allocation graph indicates:  
A) High CPU usage  
**B) Deadlock possibility**  
C) Memory shortage  
D) Race condition

**Q54.** One method for deadlock prevention is:  
A) Disallow mutual exclusion  
**B) Avoid hold and wait**  
C) Kill the process  
D) Suspend the system

**Q55.** Banker’s algorithm is used for:  
A) Deadlock detection  
B) Deadlock prevention  
**C) Deadlock avoidance**  
D) Process scheduling

**Q56.** Which strategy handles deadlock **after** it occurs?  
A) Prevention  
B) Avoidance  
**C) Detection and Recovery**  
D) None

**Q57.** Which strategy handles deadlock **after** it occurs?  
A) Prevention  
B) Avoidance  
**C) Detection and Recovery**  
D) None

**Q58.** Starvation is a situation where:  
**A) A low-priority process never gets CPU**  
B) Memory is full  
C) Deadlock occurs  
D) Process finishes quickly

**Q59.** Semaphore operations are:  
**A) Down and Up**  
B) Push and Pop  
C) Allocate and Free  
D) Get and Set

**Q60.** In bounded buffer problem, producer must wait if:  
**A) Buffer is full**  
B) Buffer is empty  
C) Consumer is inactive  
D) OS is idle

**Q61.** A Mutex is different from a semaphore because:  
**A) Semaphore allows more than one thread**  
B) Mutex is hardware-specific  
C) Mutex has no lock  
D) They are identical

**Q62.** Which of the following **can cause** deadlock?  
A) Mutual exclusion  
B) Circular wait  
C) Hold and wait  
**D) All of the above**

**Q63**. Memory management involves:  
A) Process scheduling  
B) I/O operations  
**C) Allocation and deallocation of memory**  
D) Disk partitioning

**Q63.** A **logical address** is also known as:  
A) Physical address  
**B) Virtual address**  
C) Absolute address  
D) Real address

**Q64.** Which address is generated by the CPU?  
A) Physical address  
**B) Logical address**  
C) Frame address  
D) Segment address

**Q65.** The physical address is the address:  
A) Seen by the user  
B) Used by the compiler  
**C) Used to access memory unit**  
D) Present in the process table

**Q66.** The run-time mapping from virtual to physical addresses is done by:  
A) Loader  
B) Linker  
**C) Memory Management Unit (MMU)**  
D) Compiler

**Q67.** The set of all logical addresses generated by a program is called:  
A) Physical space  
B) Logical space  
**C) Address space**  
D) Virtual memory

**Q68.** A program’s logical address space is 0 to 65535. The physical memory has 32K frames. What is the size of the address space?  
A) 32 KB  
**B) 64 KB**  
C) 128 KB  
D) 16 KB

**Q69.** The concept of **relocation** is closely related to:  
A) Static linking  
B) Memory segmentation  
**C) Physical address generation**  
D) Static scheduling

**Q70.** Address binding at **execution time** requires:  
A) Static relocation  
**B) Dynamic relocation**  
C) Linker involvement only  
D) No relocation

**Q71.** If the logical address is 1200 and the base register value is 4000, the physical address will be:  
**A) 5200**  
B) 2800  
C) 4000  
D) 1200

**Q72.** Dynamic linking refers to:  
A) Linking done at compile time  
B) Linking done at load time  
**C) Linking done during execution time**  
D) Linking of hardware and software

**Q73.** In dynamic linking, libraries are:  
A) Compiled with the application  
**B) Linked at run time**  
C) Never used  
D) Loaded permanently

**Q74.** A major advantage of dynamic linking is:  
A) Slower execution  
**B) Reduced memory usage**  
C) Higher compile time  
D) Larger binary size

**Q75.** Dynamic loading allows a routine to be:  
A) Compiled at runtime  
**B) Loaded into memory only when needed**  
C) Linked during boot  
D) Recompiled every time

**Q76.** Static linking links all libraries:  
A) At runtime  
B) At execution time  
**C) At compile time**  
D) At load time

**Q77.** Which of the following is **not** true about dynamic linking?  
A) It reduces memory footprint  
B) It improves sharing of libraries  
**C) It increases binary size**  
D) It allows late binding

**Q78.** The **dynamic linker/loader** is part of:  
A) Compiler  
**B) Operating System**  
C) Application  
D) File System

**Q79.** When a shared library is dynamically linked:  
A) Its code is copied into the program  
**B) The program loads a reference to it**  
C) It's compiled with the source  
D) It’s available at boot time

**Q80.** Dynamic linking is mostly used in:  
A) Embedded systems only  
B) Batch processing  
**C) Modern OS (e.g., Windows, Linux)**  
D) Real-time scheduling

**Q81.** DLL in Windows stands for:  
A) Dynamic Language Loader  
B) Direct Linked Library  
**C) Dynamic Link Library**  
D) Distributed Logic Library

Q.82. What is the main function of the command interpreter?  
a) to provide the interface between the API and application program  
b) to handle the files in the operating system  
**c) to get and execute the next user-specified command**  
d) none of the mentioned

Q.83. 3. In Operating Systems, which of the following is/are CPU scheduling algorithms?  
a) Priority  
b) Round Robin  
c) Shortest Job First  
**d) All of the mentioned**

Explanation: In Operating Systems, CPU scheduling algorithms are:  
i) First Come First Served scheduling  
ii) Shortest Job First scheduling  
iii) Priority scheduling  
iv) Round Robin scheduling  
v) Multilevel Queue scheduling  
vi) Multilevel Feedback Queue scheduling

Q.84. 4. To access the services of the operating system, the interface is provided by the \_\_\_\_\_  
a) Library  
**b) System calls**  
c) Assembly instructions  
d) API

Q85. CPU scheduling is the basis of \_\_\_\_\_\_\_\_\_\_\_  
**a) multiprogramming operating systems**  
b) larger memory sized systems  
c) multiprocessor systems  
d) none of the mentioned

Q86. Which one of the following is not true?  
a) kernel remains in the memory during the entire computer session  
**b) kernel is made of various modules which can not be loaded in running operating system**c) kernel is the first part of the operating system to load into memory during booting  
d) kernel is the program that constitutes the central core of the operating system

Q87. **The logical address is generated by the:**

1. MMU  
   **B) CPU**   
   C) Loader  
   D) RAM

Q88. **The actual location in memory accessed is called:**

A) Logical Address  
B) Virtual Address  
**C) Physical Address**   
D) Secondary Address

Q89. **Who is responsible for converting logical to physical address?**

A) OS Kernel  
B) Compiler  
**C) MMU**   
D) Scheduler

**Q90. Logical and physical addresses are the same in:**

A) Segmentation  
B) Paging  
**C) Compile-time binding**   
D) Runtime relocation

Q91.**Which of the following is visible to a running program?**

A) Physical address  
**B) Logical address**   
C) MMU  
D) Loader

Q92. **If base = 1000 and logical = 200, physical = ?**

**A) 1200**   
B) 1000  
C) 200  
D) Undefined

Q93. **Physical address space of a system is:**

A) Invisible to the OS  
B) Equal to logical always  
**C) Addressable by MMU**   
D) Used by compiler

Q94. **Logical Address is also called:**

A) Segment address  
**B) Virtual address**   
C) Cache address  
D) Real address

Q95. **Which one is used by hardware to access memory?**

1. Logical  
   B) Virtual  
   C) Symbolic

**D) Physical**

**Q.96 Which of the following handles relocation at runtime?**

A) Loader  
**B) MMU**   
C) Compiler  
D) Linker

Q97. Dynamic linking involves:

* A) Static libraries
* **B) Run-time linker/loader**
* C) No linking at all
* D) Hardware support only

Q98.**Which component maps logical addresses to physical addresses?**

1. ALU  
   **B. Memory Management Unit**  
   C. DMA  
   D. Interrupt Handler

Q99.**Logical and physical addresses are same in:**

1. Segmentation  
   B. Paging  
   **C. Compile time address binding**  
   D. Load time address binding

Q100.**Dynamic linking is typically used with:**

A. Static libraries  
**B. DLLs**  
C. Header files  
D. Source files

Q101.**Dynamic linking helps in:**

A. Speeding up the compilation  
**B. Reducing disk usage**  
C. Making program independent of OS  
D. Reducing runtime

Q102.**Which of the following is an example of dynamic linking?**

A. Statically embedded .lib file  
**B. Dynamically loaded .dll file at runtime**  
C. Static import of all header files  
D. None of the above

Q103.**Which address is visible to the programmer?**

A. Physical address  
B. MAC address  
**C. Logical address**  
D. IP address

**Q104.Which one is correct?**

A. Logical address = Physical address always  
B. Logical address != Physical address always  
**C. Logical address may or may not be equal to physical address**D. Logical address is same as MAC address

Q105.**In modern operating systems, the logical address space is:**

A. Identical for all processes  
B. Shared across processes  
C. **Unique for each process**  
D. Shared with the kernel

Q106. What is the main function of the kernel in an operating system?  
A. Create webpages  
**B. Manage hardware and system resources**  
C. Compile code  
D. Edit text files

107. The kernel always resides in:  
A. Hard disk  
B. User space  
**C. RAM (main memory)**  
D. Cache

108. Which of the following is **not** handled by the kernel?  
A. Process scheduling  
**B. Word formatting**  
C. Memory management  
D. I/O device control

109. **4**Which kernel type provides better modularity and fault isolation?  
A. Monolithic  
**B. Microkernel**  
C. Hybrid kernel  
D. Real-time kernel

110. **5.** Which one is used in Linux?  
A. Microkernel  
B. Hybrid  
**C. Monolithic**  
D. None of the above

111. In a microkernel, device drivers run in:  
A. Kernel space  
**B. User space**  
C. ROM  
D. Cache

**112.** Which type of kernel is used in Windows NT?  
A. Monolithic  
**B. Hybrid**  
C. Microkernel  
D. None

113. Kernel acts as an interface between:  
A. CPU and memory  
**B. Hardware and application software**  
C. Hard disk and mouse  
D. None of the above

114.The part of OS that directly communicates with the hardware is:  
A. Shell  
**B. Kernel**  
C. File system  
D. GU

115. Which is not a kernel function?  
A. Managing CPU time  
**B. Running user applications**  
C. Allocating memory  
D. Handling interrupts

116. In contiguous memory allocation, each process gets:  
A. Non-continuous memory  
**B. Continuous blocks of memory**  
C. Random memory  
D. Shared memory

117. A major problem with contiguous allocation is:  
A. Internal fragmentation  
B. Page fault  
**C. External fragmentation**  
D. Deadlock



118. Fixed partition allocation is a type of:  
A. Paging  
B. Segmentation  
**C. Contiguous allocation**  
D. Virtual memory

119. n contiguous allocation, which of the following must be known before allocation?  
A. Process ID  
B. Memory access time  
**C. Process size**  
D. OS type

120. Which is a benefit of contiguous allocation?  
A. Easy relocation  
**B. Simplicity in implementation**  
C. No fragmentation  
D. Faster compile time

121. In segmentation, memory is divided into:  
A. Fixed-sized pages  
B. Fixed-sized segments  
**C. Variable-sized segments**D. Bytes

122. A segmented address consists of:  
C**. Segment number and offset**  
D. Offset and base register

123.Segmentation helps in:  
A. Eliminating external fragmentation  
**B. Dividing memory logically (code/data/stack)**  
C. Better CPU scheduling  
D. Preventing deadlocks

124. Segment table stores:  
A. Page table entries  
B. **Segment limits and base addresses**  
C. Logical addresses  
D. CPU priorities

125. A drawback of segmentation is:  
A. Internal fragmentation  
B. Slower execution  
C**. External fragmentation**D. Too much paging

126.Paging divides memory into:  
A. Files  
**B. Frames and pages**C. Segments and offsets  
D. Variables and constants

127. Paging helps eliminate:  
A. Internal fragmentation  
**B. External fragmentation**  
C. Deadlock  
D. CPU scheduling issues

128. A page table maps:  
A. Logical to virtual addresses  
B. Virtual to hardware addresses  
**C. Pages to frames**D. Process to memory

129. What causes **internal fragmentation** in paging?  
**A. Fixed-size page and frame mismatch**  
B. Large number of segments  
C. Variable page sizes  
D. Process priority

130. What are the components of a logical address in paging?  
**A. Segment + offset**B. Page number + offset  
C. Process ID + page  
D. Frame + page

131. Page size is typically:  
**A. 1 KB to 1 MB**B. 1 Byte  
C. Same as disk block size  
D. Same as segment size

132.In paging, the physical memory is divided into:  
A. Pages  
B. Segments  
**C. Frames**D. Blocks

133. What happens during a page fault?  
A. CPU resets  
**B. Page is not in main memory and must be loaded**  
C. Process is killed  
D. Program is recompiled

134. Which of the following is stored in a page table entry?  
A. Process name  
**B. Frame number**C. Segment number  
D. Number of pages

135. Paging is a technique of:  
**A. Memory allocation**B. Process scheduling  
C. File allocation  
D. Interrupt handling

136. In segmentation with paging, each segment:  
A. Contains only one page  
B. Has no fixed size  
**C. Has its own page table**D. Cannot be paged

137. The logical address in segmentation with paging includes:  
A. Only offset  
B. Segment + frame + offset  
**C. Segment number + page number + offset**D. Page number + segment ID

138. Segmentation with paging helps in:  
A. Simplifying memory  
**B**. **Solving both internal and external fragmentation**  
C. Slowing down access  
D. Making OS single-tasking

139. In segmentation with paging, which data structure is used?  
A. One-level page table  
B. TLB  
**C. Segment table + page tables**  
D. Only segment table

140. Which of the following is **false** about segmentation with paging?  
A. Reduces fragmentation  
B. Supports modular programming  
**C. Simple hardware implementation**D. Uses two-level memory access

141. Which of the following is NOT a state in the process life cycle?  
A. Running  
B. Waiting  
**C. Paging**  
D. Ready

142. What is the role of the scheduler in an OS kernel?  
A. Manages power supply  
B. Schedules when files are saved  
**C. Decides which process runs on the CPU**D. Controls I/O devices

143. When a process is waiting for an I/O operation to complete, it is in the:  
A. Ready state  
B. Running state  
C. Terminated state  
**D. Waiting (Blocked) state**

144. Which data structure is used to store process-related information in the OS?  
A. Page Table  
B. Stack  
**C. PCB**D. FAT

145. Context switching involves:  
A. Restarting the CPU  
B. Switching from one OS to another  
**C. Saving and loading the state of processes**D. Formatting memory

146. The kernel loads a process into the CPU using:  
A. Loader  
**B. Scheduler**C. Compiler  
D. Shell

147. A process is moved from the **running** to the **ready** state when:  
A. It completes execution  
B. It is waiting for I/O  
**C. Its time slice expires**D. An interrupt occurs

148. The process control block does NOT contain:  
A. Process ID  
B. CPU register contents  
**C. File system structure**D. Process state

149.Which of the following scheduling algorithms is **preemptive**?  
A. FCFS  
B. SJF (non-preemptive)  
**C. Round Robin**  
D. Priority (non-preemptive

150. The main advantage of preemptive scheduling is:  
A. Simple implementation  
**B. Fair allocation of CPU**  
C. No context switching  
D. No interrupts

151. Which condition is essential for context switching to occur?  
A. I/O operation  
**B. Interrupt or system call**  
C. Memory overflow  
D. File open failure

152. What is the primary purpose of using **inter-process communication (IPC)**?  
A. To swap processes  
B**. To allow processes to interact or share data**C. To manage hardware  
D. To schedule tasks

153. In multiprogramming, multiple processes:  
A. Run at the exact same time  
**B. Share memory and run one after the other quickly**  
C. Run on separate CPUs only  
D. Cannot communicate

154. The time it takes for a process to switch from running to waiting and then back is called:  
A. Context change latency  
B. Response time  
C. Turnaround time  
**D. Context switching time**

**155.** Which of the following system calls is used to create a new process in Unix/Linux?  
A. exec()  
B. exit()  
C. wait()  
**D. fork()**

**156.** In First Come First Serve (FCFS) scheduling, the process that is executed first is the one:  
A. With the shortest CPU burst  
B. With the highest priority  
**C. That arrives first in the ready queue**  
D. With the largest memory size

157. FCFS scheduling is:  
A. Preemptive  
**B. Non-preemptive**C. Hybrid  
D. Real-time only

158. One major disadvantage of FCFS scheduling is:  
A. Starvation  
B. High throughput  
**C. Convoy effect**D. CPU underutilization

159. FCFS scheduling leads to **poor average waiting time** when:  
A. All processes have equal burst time  
**B. Short processes arrive after long ones**  
C. CPU is idle  
D. Processes have I/O operations

160. In FCFS, which of the following statements is TRUE?  
A. Processes are scheduled in priority order  
B. It always gives the best average waiting time  
C. Context switching is highest  
**D. No process is preempted**

**161.** In FCFS, if all processes arrive at time 0, then:  
A. Scheduling order depends on burst time  
B. Order is random  
C. Scheduling is done alphabetically  
**D. Scheduling is done based on order in the ready queue**

**162.** In FCFS, **average turnaround time** is equal to:  
A. Completion Time - Arrival Time  
B. Burst Time + Waiting Time  
**C. Both A and B**  
D. None of these

163. Which real-world example best represents FCFS scheduling?  
A. Multiple threads in a processor  
**B. Bank counter with token numbers**  
C. Elevator with shortest path optimization  
D. Round Robin sports match

164. Turnaround Time is equal to:  
A. Arrival Time – Completion Time  
**B. Burst Time + Waiting Time**C. Completion Time + Arrival Time  
D. Burst Time – Waiting Time  
✅ **Answer:** B

165. If a process has Arrival Time = 2, Burst Time = 4, Completion Time = 10, what is the Turnaround Time?  
A. 4  
B. 6  
**C. 8**D. 2

166. In which algorithm can Turnaround Time be **very high** for short processes arriving after long ones?  
A. Round Robin  
B. SJF  
**C. FCFS**  
D. SRTF

167. Which metric is used to measure the **total time taken** by a process to complete after it is submitted?  
A. Response Time  
B. Waiting Time  
**C. Turnaround Time**D. CPU Time

168. If a process completes at time 15 and arrived at time 5, its turnaround time is:  
A. 5  
**B. 10**C. 20  
D. 25

169. Turnaround time includes:  
A. Only CPU burst time  
B. Waiting time + I/O time **C. Execution time + waiting time**  
D. Only I/O time

170. If turnaround time = 12, and burst time = 8, waiting time is:  
A. 20  
**B. 4**C. 8  
D. 0

171. A scheduling algorithm that results in **lowest average turnaround time** is:  
A. FCFS  
**B. SJF (non-preemptive**)  
C. Round Robin  
D. Priority (non-preemptive)

172. Which of the following does **not affect turnaround time**?  
A. Arrival time  
B. Context switch  
C. Burst time  
**D. Program name**

173. Turnaround Time is always:  
A. Equal to Waiting Time  
B. Less than Burst Time  
**C. Greater than or equal to Burst Time**  
D. Negative

174. **The Shortest Job First algorithm schedules processes according to:**

A. Arrival Time  
B. Priority  
**C. Burst Time**  
D. Execution Time

175. **SJF algorithm gives the minimum:**

A. Turnaround Time  
B. Waiting Time  
**C. Both A and B**  
D. Response Time

176. **In non-preemptive SJF, if a new process arrives with shorter burst time during execution of a current process:**

A. It preempts the current process  
B. It goes to the front of the queue  
C**. It waits until the current process finishes**  
D. It is ignored

177. **Which scheduling algorithm suffers the most from starvation?**

A. Round Robin  
B. FCFS  
**C. SJF**D. Priority

78. **What is the main difficulty in implementing SJF scheduling?**

A. It needs special hardware  
B. Arrival times are not known  
**C. Burst times must be known in advance**  
D. High overhead

179. **If all jobs arrive at the same time in SJF, which one will execute first?**

A. The one with highest priority  
**B. The one with smallest burst time**  
C. The one with largest burst time  
D. Randomly chosen

180. **The average waiting time is minimum in which scheduling algorithm?**

A. FCFS  
**B. SJF**C. Priority  
D. Round Robin

181. **In non-preemptive SJF, when are new processes considered?**

A. Immediately upon arrival  
B. Only when CPU is free  
**C. When the running process finishes a time slice**  
D. Always after current process

182. **SJF algorithm is optimal only if:**

A. Priorities are equal  
B. Arrival times are equal  
**C. Burst times are known in advance**D. Preemption is allowed

183. **Which of the following is true for SJF non-preemptive?**

A. Processes are executed in order of arrival  
B. A running process can be preempted  
**C. Starvation can occur**D. No need to know burst times

184. **What is the biggest drawback of SJF?**

A. High overhead  
B. Inaccuracy in predicting arrival time  
**C. Starvation of longer processes**  
D. Excessive context switches

185. **What is the response time of a process in non-preemptive SJF if it starts at time 10 and arrives at time 3?**

A. 3  
**B. 7**C. 13  
D. 10

186. **Which process is selected first in SJF if multiple processes arrive at time 0?**

A. Process with highest priority  
**B. Process with shortest burst time**C. Process with longest burst time  
D. Randomly selected

187. **In SJF, if two processes have the same burst time and arrival time, which one will be selected first?**

A. Random  
B. One with lower PID  
**C. Based on OS-defined tie-breaker**D. Both at once

188. **Which scheduling algorithm reduces average waiting time most efficiently?**

**A. SJF**B. FCFS  
C. Round Robin  
D. Priority

189. **In SJF, what is the waiting time for a process that arrives after all others and has the shortest burst time?**

A. 0  
**B. It depends**  
C. Very high  
D. Cannot be determined

190**. If a long job arrives early and short jobs keep arriving later, in SJF:**

A. Long job may get delayed  
**B. Short jobs will be delayed**  
C. Long job gets priority  
D. All are executed in arrival order  
✅ **Answer:** B if SJF is Non preemptive

191. **SJF works best for:**

A. Interactive Systems  
B. Real-Time Systems  
**C. Batch Systems**  
D. Embedded Systems

192. **SJF is similar to which other algorithm when all burst times are equal?**

A. Round Robin  
B. Priority Scheduling  
**C. FCFS**D. None of the above

193. **What is the average turnaround time if waiting time is 4 units and burst time is 6 units (for a process)?**

**A. 10**B. 2  
C. 4  
D. 6

194. **In non-preemptive priority scheduling, which process is selected?**

A. First arrived process  
B. Shortest burst time process  
**C. Process with highest priority**  
D. Random process

195. **In non-preemptive priority scheduling, once a process starts:**

A. It can be interrupted by higher priority process  
**B. It finishes completely before any new process starts**C. It shares CPU in round-robin fashion  
D. It goes to ready queue again

196. **Priority scheduling may lead to:**

A. Aging  
B. Convoy effect  
**C. Starvation**D. Context switching

197. **Which is a solution to starvation in priority scheduling?**

A. Multiprocessing  
B. Round Robin  
**C. Aging**D. Blocking

198. **In priority scheduling, the lower the numeric value of priority:**

**A. The lower the actual priority**  
B. The higher the actual priority  
C. It has no effect  
D. Only affects I/O bound processes

199. **In non-preemptive priority scheduling, if two processes have the same priority, which runs first?**

A. The one with shorter burst time  
B. Random selection  
**C. The one with earlier arrival time**  
D. The one with higher process ID

200. **Which of the following statements is true for non preemptive ?**

A. Non-preemptive scheduling increases context switches  
B. Lower priority number means less importance  
C. In non-preemptive scheduling, CPU can switch any time  
**D. Once a process starts, it cannot be preempted**