一、选择题

BCCCB

二、简答题

1

抢占式调度和非抢占式调度是操作系统调度算法中的两种不同策略，它们主要区别在于进程对 CPU 的使用控制权和切换时机：

- 抢占式调度

  CPU 的使用权可以被强制收回。当一个高优先级进程到达或者当前进程运行时间过长（例如时间片用尽）时，操作系统可以中断当前进程，将其挂起，然后切换到其他进程。

  优点：能更好地响应实时需求，确保高优先级任务得到及时执行，从而提升系统的响应性和交互性。

  缺点：频繁的上下文切换会带来额外的开销，同时中断执行可能导致共享数据状态管理更复杂（需强调同步与互斥问题）。

- 非抢占式调度

  一旦进程获得 CPU 控制权，只有在该进程主动释放（例如进入等待状态、完成任务或进行 I/O 操作）时，操作系统才会调度其他进程。

  优点：上下文切换次数较少，有利于降低切换开销，也能避免因强制中断导致的共享资源冲突。

  缺点：无法快速响应紧急任务，导致高优先级任务可能长时间等待，系统响应性较差；如果进程长时间运行或进入无限循环，其他进程将得不到执行机会。

2

1. FCFS Scheduling

   Under FCFS, processes are scheduled in the order they arrive.

   Timeline:
   - At time 0.0, only P1 is available, so P1 starts execution and runs for 8 time units, finishing at time 8.0.
   - At time 0.4, P2 arrives (while P1 is executing).
   - At time 1.0, P3 arrives (while P1 is executing).
   - When P1 finishes at 8.0, both P2 and P3 are waiting. According to FCFS, P2 (which arrived earlier at 0.4) is scheduled next, running from 8.0 to 12.0 (burst = 4).
   - Finally, P3 runs from 12.0 to 13.0 (burst = 1).

   Turnaround times (finish time – arrival time):
   - P1: 8.0 – 0.0 = 8.0
   - P2: 12.0 – 0.4 = 11.6
   - P3: 13.0 – 1.0 = 12.0

   Average Turnaround Time = (8.0 + 11.6 + 12.0) / 3 ≈ 10.53

2. SJF Scheduling (Nonpreemptive)

   - At time 0.0, only P1 has arrived, so P1 is started and runs to completion by time 8.0.

- During P1′s execution, P2 (at 0.4) and P3 (at 1.0) arrive. When P1 completes at 8.0, both P2 and P3 are waiting.
- Among these, P3 has the shortest burst time (1 unit vs. 4 units for P2), so P3 is scheduled next.
- P3 runs from 8.0 to 9.0.
- Then P2 runs from 9.0 to 13.0.

Turnaround times:
- P1: 8.0 – 0.0 = 8.0
- P3: 9.0 – 1.0 = 8.0
- P2: 13.0 – 0.4 = 12.6

Average Turnaround Time = (8.0 + 8.0 + 12.6) / 3 ≈ 9.53

3. Future-Knowledge Scheduling (Idle for the First 1 Unit)

Here, we delay processing until time 1.0 to learn about all arrivals. During the idle period from 0.0 to 1.0, the processes still arrive:
- P1 arrives at 0.0
- P2 arrives at 0.4
- P3 arrives at 1.0

At time 1.0, all three processes are in the ready queue. Using nonpreemptive SJF, we select the process with the shortest burst time:
- P3 (burst 1) is scheduled first, running from 1.0 to 2.0.
- Then, with P1 (burst 8) and P2 (burst 4) remaining, we select P2 next. P2 runs from 2.0 to 6.0.
- Finally, P1 runs from 6.0 to 14.0.

Turnaround times:
- P1: 14.0 – 0.0 = 14.0
- P2: 6.0 – 0.4 = 5.6
- P3: 2.0 – 1.0 = 1.0

Average Turnaround Time = (14.0 + 5.6 + 1.0) / 3 ≈ 6.87

3

1. CPU 利用率与响应时间

优先追求高 CPU 利用率时，操作系统调度器会让 CPU 尽可能忙碌，例如采用批量处理任务或长时间运行进程占用 CPU。这可能会增加进程的等待时间，从而导致响应时间变长。

2. 平均周转时间和最长等待时间

平均周转时间侧重于整体性能表现。调度算法倾向于先处理短任务，以便降低总体平均等待时间。然而，这种策略可能会使某些长作业或低优先级作业长时间等待，从而导致最长等待时间显著增加。

3. I/O 设备利用率和 CPU 利用率

I/O 设备利用率高意味着 I/O 资源处于忙碌状态，这通常对应于 I/O 密集型任务。然而，如果调度器为保持 CPU 高利用率倾向于让 CPU 密集型进程运行，有可能出现以下冲突：
- CPU 密集型进程占据大量 CPU 时间，而 I/O 任务得不到足够的机会使用 I/O 设备，从而降低 I/O 设备的利用率。
- 为了提高 I/O 利用率，系统可能需要增加并发 I/O 任务，这可能引发更多的 CPU 等待，从而降低 CPU 的计算利用率。

这两者之间的平衡需要通过调度算法和调度策略进行权衡，以确保既不使 CPU 空闲，也不使 I/O 设备闲置，但在一定场景下，总有一个资源成为瓶颈，系统必须做出妥协。

4

1. The order is as following (P0 represents idle):
   - [000, 020): P1,
   - [020, 025): P0,
   - [025, 030): P2,
   - [030, 040): P3,
   - [040, 050): P2,
   - [050, 060): P3,
   - [060, 070): P4,
   - [070, 080): P2,
   - [080, 085): P4,
   - [085, 090): P3,
   - [090, 100): P0,
   - [100, 110): P5,
   - [110, 120): P6,
2. The turnaround time for each process is:
   - P1: 20 - 0 = 20,
   - P2: 80 - 25 = 55,
   - P3: 90 - 30 = 60,
   - P4: 85 - 60 = 25,
   - P5: 110 - 100 = 10,
   - P6: 120 - 110 = 10.
3. The waiting time for each process is:
   - P1: 20 - 20 = 0,
   - P2: 55 - 25 = 30,
   - P3: 60 - 25 = 35,
   - P4: 25 - 15 = 10,
   - P5: 10 - 10 = 0,
   - P6: 10 - 10 = 0.
4. The CPU utilization rate is:

   CPU Utilization = (Total CPU Time) / (Total Time) = (120 - 5 - 10) / (120) = 87.5%