--Question Starting--
3. In a memory management system employing paging with demand paging and a least recently used (LRU) page replacement policy, consider the following statements:
I. Increasing the size of a page reduces the number of page faults in a program with localized reference behavior.
II. Throttling the page replacement algorithm to favor pages with higher reference frequency minimizes the probability of page thrashing.
III. When the working set of a process exceeds the total number of available frames, the system experiences thrashing, leading to significant performance degradation.
Which of the following is correct?
(1) I only
(2) II only
(3) I and III only
(4) All of the above
Answer Key: 4
Solution:
? Statement I(Correct): Larger page sizes mean fewer pages, potentially reducing page faults if the program exhibits locality. Less frequent page loading occurs because each page contains more data, thereby reducing the number of page faults for localized reference patterns.
? Statement II(Correct): Prioritizing pages with higher reference frequency in the replacement policy ensures that frequently used pages stay in memory longer, minimizing the chance of replacing useful pages, which helps in reducing thrashing.
? Statement III(Correct): When the working set exceeds the available frames, the system continually swaps pages in and out, causing thrashing. This results in excessive disk I/O and degraded performance.
Hence, Option (4) is the right answer.

--Question Starting--
4. Consider a relational database schema where normalization is performed to remove redundancy and update anomalies. Suppose functional dependencies are given as: A ? B, B ? C, and A ? D. The relation R is initially unnormalized with attributes (A, B, C, D, E). Which of the following statements accurately describe the normalization process and its implications?
I. Applying 2NF to R ensures that all non-prime attributes are fully functionally dependent on the primary key.
II. Decomposition of R into Boyce-Codd Normal Form (BCNF) guarantees elimination of all anomalies related to functional dependencies.
III. The presence of transitive dependencies, such as A ? B and B ? C, necessitates normalization to at least 3NF.
Which of the following is correct?
(1) I and II only
(2) II only
(3) I and III only
(4) All of the above
Answer Key: 2
Solution:
? Statement I(Incorrect): 2NF deals with removing partial dependencies; however, without specifying candidate keys, it is uncertain whether attributes like D are fully dependent on the primary key. The focus here is on partial dependency, but the statement simplifies the concept.
? Statement II(Correct): BCNF is stricter than 3NF, eliminating all functional dependencies except those that are trivial, thus ensuring anomalies are removed.
? Statement III(Incorrect): Transitive dependencies (like A ? B and B ? C) are addressed in 3NF, but normalization beyond 3NF is required for certain anomalies; BCNF further refines this.
Therefore, only statement II correctly describes the normalization process implications.
Hence, Option (2) is the right answer.

--Question Starting--
5. In the context of algorithms for polynomial multiplication, consider the following statements:
I. The naive polynomial multiplication algorithm has a time complexity of $O(n^2)$, where n is the degree of the polynomials.
II. Fast Fourier Transform (FFT) based polynomial multiplication reduces the time complexity to $O(n \log n)$.
III. To multiply two polynomials using FFT, their degrees must be padded to the next power of two to facilitate efficient computation.
Which of the following is correct?
(1) I only
(2) I and II only
(3) II and III only
(4) All of the above
Answer Key: 4
Solution:
? Statement I(Correct): The naive approach multiplies each coefficient of one polynomial with every coefficient of the other, resulting in $O(n^2)$ complexity.
? Statement II(Correct): FFT-based polynomial multiplication leverages polynomial evaluation and interpolation, reducing complexity to $O(n \log n)$.
? Statement III(Correct): FFT algorithms are most efficient when data sizes are powers of two; hence, polynomials are padded to the next power of two to prevent computational inefficiencies.
All three statements are accurate descriptions of polynomial multiplication algorithms and their properties.
Hence, Option (4) is the right answer.