

Indian Institute of Technology, Kharagpur

Class Test - 2

Date of Examination: 28-03-2023 Session: FN (9-9.45 AM) Duration: 45 mins
 Subject No.: CS30202 Subject: DATABASE MANAGEMENT SYSTEM
 Department/Center/School: Computer Science and Engineering
 Specific charts, graph paper, log book etc., required: NO Total Marks : 20
 Special instructions (if any): ANSWER ALL QUESTIONS
 Note: All parts of the question (a,b,c,...) should be answered at a stretch.

1. Answer the following in brief:

(2+2 = 4)

- (a) Consider the following data and parity blocks arrangement on six disks using RAID level 5 arrangement with block-level striping. In the below matrix (2-D array), each column represents a disk. The D_i s represent data blocks; the P_i s represent parity blocks. Parity block P_i is derived from data blocks $D_{6i-5}D_{6i-2}$ to D_{6i} .

- i. What's wrong with the arrangement of data and parity blocks as shown in the below matrix?
- ii. What are the issues that arise out of the arrangement of data and parity blocks?

$D1$	$D2$	$D3$	$D4$	$D5$	$D6$
$P1$	$D7$	$D8$	$D9$	$D10$	$D11$
$D12$	$P2$	$D13$	$D14$	$D15$	$D16$
.
.
.
.

- (b) In view of database buffer management, explain the functionality of Pin and Unpin operations.

2. In view of B^+ tree, answer the following in brief: (2+4+10 = 16)

- (a) With appropriate diagrams discuss the salient features of internal and leaf nodes.
- (b) Compute the number of keys present in the internal node and leaf node for the following scenario: Size of the block = 512 bytes, size of the key = 8 bytes, size of the node-pointer = 6 bytes and size of the record pointer = 8 bytes. Assume that each of the the internal node or leaf node will fit into one block.
- (c) Construct a B^+ tree for the order 3 (i.e., 3 pointers and 2 keys) with the following key values inserted in the order as shown below : 20,11,14,25,30,12,22,23,24. Assume that the tree is initially empty. Show the construction of B^+ tree incrementally for each key insertion and indicate all events (such as overflow, splitting, etc..) at all levels of the tree, for each key insertion.