

# COMPETISHUN

## CBSE Class 12 — Computer Science (Code 083)

Sample Paper: Questions and Solutions — Session 2026-27

**Time: 3 hours    Max. Marks: 70    Questions: 37    Sections: 5**

Section A — Q1–21 (1 mark each, 21) | Section B — Q22–28 (2 marks each, 14) | Section C — Q29–31 (3 marks each, 9) | Section D — Q32–35 (4 marks each, 16) | Section E — Q36–37 (5 marks each, 10)

All questions are compulsory; internal choices are provided in some questions — attempt only one of the choices. All programming answers use Python only. In an MCQ, the text of the correct option must also be written.

### Section A

Questions 1 to 21 · 1 mark each · 21 marks

#### Question 1 — 1 Mark

Write the output on execution of the following Python code:

```
L1 = [10, 20, 25, 30]
L1.insert(1, 15)
print(L1)
```

**Answer** [10, 15, 20, 25, 30]

**Explanation** insert(1, 15) places 15 at index 1 and shifts every later element one position to the right.

**Marking Scheme** 1 mark for the correct answer.

#### Question 2 — 1 Mark

What will be the output of the following expression?

```
Msg = "programming is amazing"
print(Msg.split("m"))
```

- a) ['progra', '', 'ing is a', 'azing']      b) ['progra', 'ming is a', 'azing']  
c) ('progra', '', 'ing is amazing')      d) ['program', 'ing is a', 'zing']

**Answer** a) ['progra', '', 'ing is a', 'azing']

**Explanation** The string contains three m characters. Splitting on m gives four pieces; the two consecutive ms in “programming” produce an empty string between them. split() always returns a list, so option (c) is ruled out.

**Marking Scheme** 1 mark for the correct answer.

#### Question 3 — 1 Mark

Which of the following data types in Python is mutable?

- a) Tuple      b) String  
c) Integer      d) List

**Answer** d) List

**Explanation** A list can be changed in place after creation; tuples, strings and integers are immutable.

**Marking Scheme** 1 mark for the correct answer.

#### Question 4 — 1 Mark

A column in one table matching a primary key of another table is called a:

- a) Composite key      b) Candidate key  
c) Foreign key      d) Super key



**Question 8 — 1 Mark**

Long-distance communication without line-of-sight is mainly achieved using:

- a) Microwaves
- b) Ultraviolet
- c) Radio waves
- d) X-rays

**Answer** c) Radio waves

**Explanation** Radio waves are omnidirectional and can be reflected by the ionosphere and bend around obstacles, so they travel long distances without a clear line of sight. Microwaves need line-of-sight transmission.

**Marking Scheme** 1 mark for the correct answer.

**Question 9 — 1 Mark**

Which MySQL query will display all employees whose salaries are between 90,000 and 100,000 (including limits)?

- a) `SELECT * FROM Employees WHERE salary > 90000 AND salary < 100000;`
- b) `SELECT * FROM Employees WHERE salary IN (90000, 100000);`
- c) `SELECT * FROM Employees WHERE salary BETWEEN 90000 AND 100000;`
- d) `SELECT * FROM Employees WHERE salary = 90000 OR salary = 100000;`

**Answer** c) `SELECT * FROM Employees WHERE salary BETWEEN 90000 AND 100000;`

**Explanation** BETWEEN ... AND ... is inclusive of both limits. Option (a) excludes the limits; options (b) and (d) match only the two exact values.

**Marking Scheme** 1 mark for the correct answer.

**Question 10 — 1 Mark**

What will be the output of the following Python statement:

```
print(25 + 100 ** 1 ** 3 / 5)
```

**Answer** 45.0

**Explanation** \*\* is right-associative, so  $1^{**}3$  is evaluated first and gives 1; then  $100^{**}1$  is 100;  $100/5$  is 20.0; finally  $25 + 20.0$  is 45.0. Division always returns a float.

**Marking Scheme** 1 mark for the correct answer.

**Question 11 — 1 Mark**

What is the data type of the result when the expression `5 // 2` is evaluated in Python?

- a) float
- b) int
- c) string
- d) None

**Answer** b) int

**Explanation** // is floor division. With both operands of type int the result is int — here 2.

**Marking Scheme** 1 mark for the correct answer.

**Question 12 — 1 Mark**

Every network interface card (NIC) comes with its own \_\_\_\_ address.

- a) Internet Protocol (IP)
- b) Open Systems Interconnection (OSI)
- c) Physical (MAC)
- d) Dynamic Host Configuration Protocol (DHCP)

**Answer c) Physical (MAC)**

**Explanation** The MAC (physical) address is burnt into the NIC by the manufacturer and is globally unique; an IP address, by contrast, is assigned by the network.

**Marking Scheme** 1 mark for the correct answer.

**Question 13 — 1 Mark**

In a student table, "Class" can only take values from {IX, X, XI, XII}. This set is called the \_\_\_\_.

- a) Domain
- b) Tuple
- c) Attribute
- d) Key

**Answer a) Domain**

**Explanation** The domain of an attribute is the set of all permissible values that the attribute may take.

**Marking Scheme** 1 mark for the correct answer.

**Question 14 — 1 Mark**

What would be written in the file errorlog.txt after the following code is run?

```
def data(num):  
    try:  
        return 190 / (29 - num)  
    except ZeroDivisionError:  
        logfile = open("errorlog.txt", "w")  
        logfile.write("Num was 29")  
        logfile.close()  
  
data(29)
```

**Answer Num was 29**

**Explanation** data(29) makes the denominator  $29 - 29 = 0$ , which raises ZeroDivisionError. The handler opens the file in "w" mode and writes the text Num was 29.

**Marking Scheme** 1 mark for the correct answer.

**Question 15 — 1 Mark**

Which of the following will NOT change if you insert a new record into a table?

- a) Cardinality
- b) Degree
- c) Number of tuples
- d) Total rows

**Answer b) Degree**

**Explanation** Degree is the number of attributes (columns) in a relation and is unaffected by inserting a row. Cardinality, number of tuples and total rows all mean the row count, which increases by one.

**Marking Scheme** 1 mark for the correct answer.

**Question 16 — 1 Mark**

What will be the output of the following Python code?

```
y = 8  
  
def update():  
    y = y + 2  
    print(y)  
  
update()
```

- a) 10
- b) 8

c) Error

d) None

**Answer c) Error**

**Explanation** Assigning to `y` inside `update()` makes `y` a local name for the whole function, so reading it in `y = y + 2` before assignment raises `UnboundLocalError`. Declaring `global y` would fix it.

**Marking Scheme** 1 mark for the correct answer.

### Question 17 — 1 Mark

In \_\_\_ switching, before a communication starts, a dedicated path is identified between the sender and the receiver.

a) Graph

b) Plot

c) Circuit

d) Packet

**Answer c) Circuit**

**Explanation** In circuit switching a physical path is reserved for the entire duration of the communication, as in a traditional telephone call.

**Marking Scheme** 1 mark for the correct answer.

### Question 18 — 1 Mark

What will be the output of the following Python code?

```
text = "Drone Expo"
print(text[-2::-4])
```

**Answer peD**

**Explanation** In "Drone Expo", index -2 is p (index 8). A step of -4 then moves to index 4 (e) and index 0 (D), giving peD.

**Marking Scheme** 1 mark for the correct answer.

### Question 19 — 1 Mark

Which MySQL command correctly removes an attribute Age from table Student?

a) ALTER TABLE Student DROP COLUMN Age;

b) ALTER TABLE Student REMOVE Age;

c) DROP COLUMN Age FROM Student;

d) DELETE COLUMN Age FROM Student;

**Answer a) ALTER TABLE Student DROP COLUMN Age;**

**Explanation** Removing a column changes the \*structure\* of the table, so the DDL command ALTER TABLE ... DROP COLUMN is used. DELETE removes rows, not columns.

**Marking Scheme** 1 mark for the correct answer.

### Questions 20 and 21 — Assertion (A) and Reason (R)

**Q20 and Q21 are Assertion (A) and Reason (R) based questions. Mark the correct choice as:**

a) Both A and R are True and R is the correct explanation for A.

b) Both A and R are True and R is not the correct explanation for A.

c) A is True but R is False.

d) A is False but R is True.

**Question 20 — 1 Mark**

**Assertion (A):** Opening a file in mode `rb` allows you to read a binary file without modifying it.

**Reason (R):** In `rb` mode, if the file does not exist, Python creates a new empty file.

**Answer c) A is True but R is False.**

**Explanation** A is true — `rb` opens an existing binary file for reading only, so its contents cannot be altered. R is false — in any read mode (`r`, `rb`) a missing file raises `FileNotFoundError`; it is the write and append modes (`w`, `wb`, `a`, `ab`) that create a new file.

**Marking Scheme** 1 mark for the correct answer.

**Question 21 — 1 Mark**

**Assertion (A):** The MySQL query `SELECT Student_Name FROM Student WHERE Guardian_Phone IS NULL`; displays the names of students whose guardian phone number is not recorded.

**Reason (R):** In MySQL, you can also use `=` to check for NULL values; it is equivalent to `IS NULL` to identify columns with no value.

**Answer c) A is True but R is False.**

**Explanation** A is true — `IS NULL` is the correct way to find rows where a value has not been entered. R is false — `NULL` means “unknown”, so a comparison such as `Guardian_Phone = NULL` evaluates to `NULL` (never `TRUE`) and returns no rows. Only `IS NULL` / `IS NOT NULL` work.

**Marking Scheme** 1 mark for the correct answer.

**Section B**

Questions 22 to 28 · 2 marks each · 14 marks

**Question 22 — 2 Marks**

A. Define global variable and local variable with a suitable example.

OR

B. Write the difference between the `seek()` and `tell()` methods in Python text file handling with a suitable example.

**Answer and Solution — A**

**Global variable** — declared outside every function; it is accessible from anywhere in the program.

```
x = 10          # global variable
print(x)
```

**Local variable** — declared inside a function; it exists only while that function runs and is accessible only inside it.

```
def demo():
    y = 5        # local variable
    print(y)
```

**Answer and Solution — B**

`tell()` returns the current position (in bytes) of the file pointer, whereas `seek(offset)` moves the file pointer to the given position.

```
f = open("file.txt", "r")
print(f.tell())    # e.g. 0 - pointer at the beginning
f.seek(5)          # moves the pointer to the 5th byte
print(f.tell())    # 5
```

**Marking Scheme** 1 mark for the correct difference / definitions +  $\frac{1}{2}$  mark for each correct example = **2 marks**.

### Question 23 — 2 Marks

The Python program below tries to print the list in reverse. However, there are syntax and logical errors in the code. Rewrite it after removing all the errors. Also, underline all the corrections made.

```
words=["Class Test","Term Exams","Pre-board Exam","Board Exam']
L=len(words)
for i in range(L, -1, -1)
    Print (words[i] , end = " ")
```

#### Answer and Solution — corrected program (corrections underlined>

```
words=["Class Test","Term Exams","Pre-board Exam","Board Exam_"]
L=len(words)
for i in range(L-1, -1, -1):
    print (words[i] , end = " ")
```

#### Errors corrected

- The closing quote of "Board Exam'" was a typographic quote — replaced by a straight double quote.
- A colon was missing at the end of the for statement header.
- Print must be written in lower case as print — Python is case-sensitive.
- range(L, -1, -1) starts at index L, which is out of range — it must start at L-1.

**Marking Scheme** ½ mark for each of the 4 corrections = **2 marks**.

### Question 24 — 2 Marks

**A.** Write a Python function `capitalize_words(words)` that takes a list of strings `words` and returns a new list with each word capitalized.

**Example:** if the input is ["python", "computer", "science"], the output should be ["Python", "Computer", "Science"].

OR

**B.** Write a Python function `sum_positive(list1)` that takes a list of numbers and prints the sum of all positive numbers in the list.

**Example:** if the input is [10, -5, 20, -3, 7], the output should be 37.

#### Answer and Solution — A

```
def capitalize_words(words):      # words is a list of strings
    result = []
    for w in words:
        result.append(w.capitalize())
    return result

print(capitalize_words(["python", "computer", "science"]))
```

#### Answer and Solution — B

```
def sum_positive(list1):          # list1 is a list of numbers
    total = 0
    for num in list1:
        if num > 0:
            total += num
    print(total)

sum_positive([10, -5, 20, -3, 7])
```

**Marking Scheme** ½ mark for the function definition + 1 mark for the correct / similar logic + ½ mark for the correct return or print statement = **2 marks**.

### Question 25 — 2 Marks

**A.** Answer using Python built-in methods / functions only:

I. Write a statement to check whether the string `username` starts with "admin".

II. Given data = [2, 7, 9, 9, 2], write a statement to count how many times the largest element occurs.

OR

B. Predict the output of the following Python code:

```
text = "Python Programming"
print(text.replace("o", "0"))
print(text.find("z"))
```

**Answer and Solution — A**

I. username.startswith("admin")

II. data.count(max(data)) # max(data) is 9, which occurs 2 times

**Answer and Solution — B**

```
Pyth0n Pr0gramming
-1
```

replace() substitutes every occurrence of "o"; find() returns -1 when the substring is absent (it does not raise an error, unlike index()).

**Marking Scheme** 1 mark for each correct statement / output line = **2 marks**.

### Question 26 — 2 Marks

Predict the output of the following Python code:

```
code = "gLaD5"
result = []
i = 0
while i < len(code):
    if code[i].isupper():
        result.append(code[i].lower())
    elif code[i].isdigit():
        result.append(int(code[i]) * 2)
    else:
        result.append("#")
    i += 1
print(result)
```

**Answer** ['#', 'l', '#', 'd', 10]

**Solution — character by character**

| Index | Character | Test satisfied           | Appended          |
|-------|-----------|--------------------------|-------------------|
| 0     | g         | none (lower-case letter) | '#'               |
| 1     | L         | isupper()                | 'l'               |
| 2     | a         | none (lower-case letter) | '#'               |
| 3     | D         | isupper()                | 'd'               |
| 4     | 5         | isdigit()                | 10 (int('5') * 2) |

**Marking Scheme** **2 marks** for the fully correct output.

### Question 27 — 2 Marks

A. Write suitable commands to do the following in MySQL.

- I. Display a list of all the relations stored in a database.
- II. Delete a database named CS\_SP.

OR

**B. Differentiate between the ALTER and UPDATE commands in MySQL with a suitable example.**

**Answer and Solution — A**

- I. SHOW TABLES;  
II. DROP DATABASE CS\_SP;

**Answer and Solution — B**

| ALTER                                                                                                  | UPDATE                                                                                            |
|--------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| A DDL (Data Definition Language) command.                                                              | A DML (Data Manipulation Language) command.                                                       |
| Changes the structure (schema) of a table — adds, modifies, renames or drops a column or a constraint. | Changes the data stored in the existing rows of a table — modifies values in one or more columns. |
| Example: ALTER TABLE Student ADD Age INT;                                                              | Example: UPDATE Student SET Age = 17 WHERE RollNo = 5;                                            |

**Marking Scheme** **A** — 1 mark for each correct command. **B** — 1 mark for the correct difference + ½ mark for each correct example = **2 marks**.

### Question 28 — 2 Marks

**A. Which protocol helps us to browse through web pages using internet browsers? Name any one internet browser.**

OR

**B. Define tree topology. Also, write down any one limitation of tree topology.**

**Answer and Solution — A**

- **HTTP (HyperText Transfer Protocol)** is the protocol used to browse web pages through internet browsers. (Its secure version is HTTPS.)
- Any one browser: **Google Chrome / Mozilla Firefox / Microsoft Edge / Safari / Opera.**

**Answer and Solution — B**

- **Tree topology** combines the characteristics of bus and star topologies: groups of star-configured computers are connected to a central bus backbone.
- **Limitation (any one)** — failure of the main backbone cable brings down the entire network; it is expensive because it needs a large amount of cable and many network devices; it is difficult to configure and maintain as the network grows.

**Marking Scheme** **A** — 1 mark for the correct protocol + 1 mark for any one browser name. **B** — 1 mark for the correct definition + 1 mark for any one limitation = **2 marks**.

## Section C

Questions 29 to 31 · 3 marks each · 9 marks

### Question 29 — 3 Marks

**A. Write and call a Python function to read "History.txt" and display all lines that contain a four-digit year (e.g. 1999, 2024 etc.).**

OR

**B. Write a Python function that displays the number of times the word "Python" appears in a text file named "Prog.txt".**

**Answer and Solution — A**

```
def DisplayYearLines():
    f = open("History.txt", "r")
    for line in f:
        words = line.split()
        for w in words:
            if w.isdigit() and len(w) == 4:
                print(line)
                break
    f.close()

DisplayYearLines()
```

**Marking Scheme** ½ mark for the correct function header + ½ mark for correctly opening the file + 1½ marks for the correct / similar logic for displaying the desired lines + ½ mark for correctly calling the function = **3 marks**.

**Answer and Solution — B**

```
def count_python():
    count = 0
    with open("Prog.txt", "r") as file:
        text = file.read()
    words = text.split()
    for word in words:
        if word.lower() == "python":
            count += 1
    print("The word Python appears", count, "times.")

count_python()
```

**Marking Scheme** ½ mark each for — the function header, opening the file, reading from the file, splitting the text into words, correct use of the counter variable, and displaying the result = **3 marks**.

**Question 30 — 3 Marks**

A list containing records of books is given as:

```
B = [("Machine Learning Basics", 550), ("World Geography", 450),
      ("Wonder World", 140), ("AI Revolution", 750)]
```

Write the following user-defined functions to perform operations on a stack named Books:

- I. Push\_element() : push only those books whose price is more than 250 into the stack.
- II. Pop\_element() : pop the items from the stack and display them. After all items are popped, display "Stack Empty".

**Expected Output:**

```
('AI Revolution', 750)
('World Geography', 450)
('Machine Learning Basics', 550)
Stack Empty
```

**Answer and Solution**

```

B = [("Machine Learning Basics", 550), ("World Geography", 450),
      ("Wonder World", 140), ("AI Revolution", 750)]
Books = []

# I.
def Push_element():
    for item in B:
        if item[1] > 250:
            Books.append(item)

# II.
def Pop_element():
    while Books:
        print(Books.pop())
    print("Stack Empty")

Push_element()
Pop_element()

```

Only three books cost more than 250, so the stack holds them in insertion order; pop() removes from the top (the end of the list), which produces the expected LIFO output.

**Marking Scheme** 1½ marks for each correct function = **3 marks**.

**Question 31 – 3 Marks**

**A.** Write a Python function avg\_len() that adds up the lengths of all the words in a list and then prints the average (mean) length.

OR

**B.** Consider the tuple pairs = ((2,5), (4,2), (9,8), (12,10)). Write a Python function cnt\_even() to count the number of pairs (a, b) where both a and b are even.

**Answer and Solution – A**

```

def avg_len(words):
    total = 0
    count = 0
    for w in words:
        total += len(w)
        count += 1
    return total / count

my_words = ["computer", "science", "python", "data"]
print("Average word length:", avg_len(my_words))    # 6.25

```

**Answer and Solution – B**

```

pairs = ((2,5), (4,2), (9,8), (12,10))

def cnt_even(t):
    c = 0
    for a, b in t:
        if a % 2 == 0 and b % 2 == 0:
            c += 1
    return c

print("Number of even-even pairs:", cnt_even(pairs))    # 2

```

Only (4, 2) and (12, 10) have both members even, so the count is 2.

**Marking Scheme** ½ mark for the function definition + 2 marks for the correct / similar logic + ½ mark for the print / return statement = **3 marks**.

## Section D

Questions 32 to 35 · 4 marks each · 16 marks

### Question 32 — 4 Marks

Consider the table TECH\_EVENT\_REG as given below:

| reg_id | participant_name | event_name       | grade | reg_fee |
|--------|------------------|------------------|-------|---------|
| 501    | Aarav            | Hackathon        | 11    | 200     |
| 502    | Riya             | RoboRace         | 10    | 150     |
| 503    | Krish            | WebDev Challenge | 12    | 250     |
| 504    | Navya            | Hackathon        | 11    | 200     |
| 505    | Samaira          | CodeSprint       | 9     | 100     |
| 506    | Rohan            | RoboRace         | 10    | 150     |

A. Write the following queries:

- To display the number of participants in each event where the number of participants is greater than 1.
- To display the distinct event names available in the Tech Event.
- To display details of participants whose names contain the letters 'an'.
- To display all records of TECH\_EVENT\_REG sorted by participant\_name in descending order.

OR

B. Predict the output of the following:

- Select participant\_name, event\_name from tech\_event\_reg where grade = 11;
- Select event\_name, count(\*) from tech\_event\_reg group by event\_name;
- Select participant\_name from tech\_event\_reg where event\_name like '%challenge';
- Select max(reg\_fee) from tech\_event\_reg where grade in (9, 10);

Answer and Solution — A

- ```
SELECT event_name, COUNT(*) AS total_participants
FROM tech_event_reg
GROUP BY event_name
HAVING COUNT(*) > 1;
```
- ```
SELECT DISTINCT event_name FROM tech_event_reg;
```
- ```
SELECT * FROM tech_event_reg WHERE participant_name LIKE '%an%';
```
- ```
SELECT * FROM tech_event_reg ORDER BY participant_name DESC;
```

Answer and Solution — B

I.

| participant_name | event_name |
|------------------|------------|
| Aarav            | Hackathon  |
| Navya            | Hackathon  |

II.

| event_name       | COUNT(*) |
|------------------|----------|
| Hackathon        | 2        |
| RoboRace         | 2        |
| WebDev Challenge | 1        |

| event_name | COUNT(*) |
|------------|----------|
| CodeSprint | 1        |

**III.**

| participant_name |
|------------------|
| Krish            |

**IV.**

| MAX(reg_fee) |
|--------------|
| 150          |

For III, '%challenge' matches a value that \*ends\* with "challenge"; MySQL string comparison is case-insensitive by default, so WebDev Challenge qualifies. For IV, grades 9 and 10 give fees 100, 150 and 150, so the maximum is 150.

**Marking Scheme** 4 × 1 mark for each correct query (part A) or each correct output (part B) = **4 marks**.

**Question 33 — 4 Marks**

Kabir runs a vehicle rental agency and maintains a CSV file named `Rentals.csv`. The file stores daily rental details with columns `Vehicle_ID`, `Vehicle_Type`, `Hours_Rented`, `Rate_Per_Hour`. Create the following user-defined functions:

- I. `AddRental()` — to accept rental details from the user and append them to `Rentals.csv`.
- II. `CalculateTotalRent()` — to calculate and return the total rent earned (`Hours_Rented × Rate_Per_Hour`).

**Answer and Solution**

```
import csv

# I.
def AddRental():
    Vehicle_ID = input("Enter Vehicle ID: ")
    Vehicle_Type = input("Enter Vehicle Type: ")
    Hours_Rented = input("Enter Hours Rented: ")
    Rate_Per_Hour = input("Enter Rate Per Hour: ")
    with open("Rentals.csv", "a") as f:
        w = csv.writer(f)
        w.writerow([Vehicle_ID, Vehicle_Type, Hours_Rented, Rate_Per_Hour])
    print("Rental record added successfully.")

# II.
def CalculateTotalRent():
    total = 0
    f = open("Rentals.csv", "r")
    r = csv.reader(f)
    for row in r:
        hours = int(row[2])
        rate = int(row[3])
        total += hours * rate
    f.close()
    print("Total Rent Earned:", total)
    return total
```

*Note: `import csv` may be treated as part of the complete program and need not be repeated in each part.*

**Marking Scheme** **I** — ½ mark each for correctly taking user input, opening the file in append mode, creating the writer object, and using `writerow()` = 2 marks. **II** — ½ mark each for opening the file in the right mode, creating the reader object, correct calculation after converting to integers, and correctly returning / displaying the total rent = 2 marks. **Total 4 marks.**

### Question 34 — 4 Marks

Aditi is managing an events database and needs to access certain information from the Chess\_Event and Registrations tables for an upcoming survey. Help her extract the required information by writing the appropriate MySQL queries as per the tasks mentioned below:

**Table: Chess\_Event**

| Event_ID | Event_Name                      | Event_Date | Venue                   | Max_Participants |
|----------|---------------------------------|------------|-------------------------|------------------|
| 101      | Inter-School Chess Championship | 2025-01-15 | Auditorium Hall         | 60               |
| 102      | Rapid Chess Tournament          | 2025-02-10 | Activity Room 3         | 40               |
| 103      | Under-16 Chess Cup              | 2025-03-05 | Sports Complex Room 2   | 50               |
| 104      | Teachers vs Students Chess Meet | 2025-04-12 | Library Conference Area | 30               |
| 105      | Open Blitz Chess Challenge      | 2025-05-20 | IT Lab 1                | 80               |

**Table: Registrations**

| Reg_ID | Student_Name     | Class | Event_ID | Registration_Date |
|--------|------------------|-------|----------|-------------------|
| 1      | Aarav Singh      | 9A    | 101      | 2024-12-20        |
| 2      | Navya Sharma     | 9C    | 103      | 2025-01-12        |
| 3      | Rudransh Verma   | 10B   | 102      | 2025-01-25        |
| 4      | Daksh Lal        | 9B    | 104      | 2025-02-18        |
| 5      | Abhijay Malhotra | 11A   | 105      | 2025-03-30        |
| 6      | Krish Sharma     | 10A   | 101      | 2024-12-22        |
| 7      | Manas Taneja     | 8C    | 102      | 2025-01-28        |
| 8      | Riya Kapoor      | 11B   | 105      | 2025-04-01        |
| 9      | Harsh Mehta      | 12A   | 103      | 2025-02-20        |
| 10     | Simran Kaur      | 9D    | 104      | 2025-03-05        |

- I. Write a MySQL query to update the venue of the event “Inter-School Chess Championship” to “Multi-Purpose Hall”.
- II. Display the details of all chess events scheduled after 2025-01-01.
- III. Write a MySQL query to display the registration details of students who have registered for events scheduled in ‘Auditorium Hall’, ‘Activity Room 3’, or ‘Library Conference Area’.
- IV. (a) Display the Student\_Name along with the Event\_Name for all registrations, **OR** (b) write a MySQL query to display the cartesian product of the Chess\_Event and Registrations tables.

#### Answer and Solution

- I. 

```
UPDATE Chess_Event
SET Venue = 'Multi-Purpose Hall'
WHERE Event_Name = 'Inter-School Chess Championship';
```
- II. 

```
SELECT * FROM Chess_Event WHERE Event_Date > '2025-01-01';
```
- III. 

```
SELECT R.* FROM Registrations R JOIN Chess_Event C
ON R.Event_ID = C.Event_ID
WHERE C.Venue IN ('Auditorium Hall', 'Activity Room 3',
'Library Conference Area');
```
- IV. (a) 

```
SELECT R.Student_Name, C.Event_Name
FROM Registrations R JOIN Chess_Event C
ON R.Event_ID = C.Event_ID;
```

  
 (b) 

```
SELECT * FROM Chess_Event, Registrations;
```

**Marking Scheme**  $4 \times 1$  mark for each correct query = **4 marks**. For IV, either alternative (a) or (b) is to be attempted.

### Question 35 — 4 Marks

A MySQL database named LibraryDB has a table book\_records which contains the following attributes:

- Book\_ID — ID of the book (Integer)
- Book\_Title — title of the book (String)
- Author — author of the book (String)
- Copies — number of copies available (Integer)

Consider the following details to establish Python-MySQL connectivity — **Username:** librarian, **Password:** lib@2025, **Host:** localhost.

Write a Python-MySQL program to change the Book\_Title to 'Data Structures in Python' for the record whose Book\_ID is 109 in the book\_records table.

#### Answer and Solution

```
import mysql.connector

con = mysql.connector.connect(host="localhost", user="librarian",
                             password="lib@2025", database="LibraryDB")

cur = con.cursor()

q = "UPDATE book_records SET Book_Title='Data Structures in Python' \
    WHERE Book_ID=109"
cur.execute(q)
con.commit()

print("Book title updated successfully.")
cur.close()
con.close()
```

commit() is essential — without it the UPDATE is not saved permanently in the database.

**Marking Scheme**  $\frac{1}{2}$  mark for correctly importing the connector +  $\frac{1}{2}$  mark for creating the connection object +  $\frac{1}{2}$  mark for creating the cursor object + 1 mark for the correct UPDATE query + 1 mark for executing the query with commit() +  $\frac{1}{2}$  mark for correctly closing the connection = **4 marks**.

## Section E

Questions 36 and 37 · 5 marks each · 10 marks

### Question 36 — 2 + 3 = 5 Marks

A college administrator, Atul, maintains student course enrollment records. Each record includes Roll\_No, Student\_Name, Course, Fees. Write Python functions to:

- Input enrollment details and append them to a binary file. [2]
- Update the Fees of all students enrolled in the "Data Science" course to 75000. [3]

#### Answer and Solution — I

```
import pickle

def AddEnrollment():
    Roll = int(input("Enter Roll No: "))
    Name = input("Enter Student Name: ")
    Course = input("Enter Course: ")
    Fees = int(input("Enter Fees: "))
    rec = [Roll, Name, Course, Fees] # record as a list
    with open("Enrollments.dat", "ab") as f:
        pickle.dump(rec, f)
    print("Record added successfully.")
```

**Marking Scheme** ½ mark each for — correctly defining the function header, opening the file in append mode, taking user input, and using the dump ( ) method of the pickle module = **2 marks**.

**Answer and Solution — II**

```
def UpdateFees():
    updated = False
    records = []
    with open("Enrollments.dat", "rb") as file:
        try:
            while True:
                rec = pickle.load(file)
                if rec[2] == "Data Science":    # Course is at index 2
                    rec[3] = 75000             # Fees is at index 3
                    updated = True
                    records.append(rec)
            except EOFError:
                pass

    with open("Enrollments.dat", "wb") as file:
        for rec in records:
            pickle.dump(rec, file)

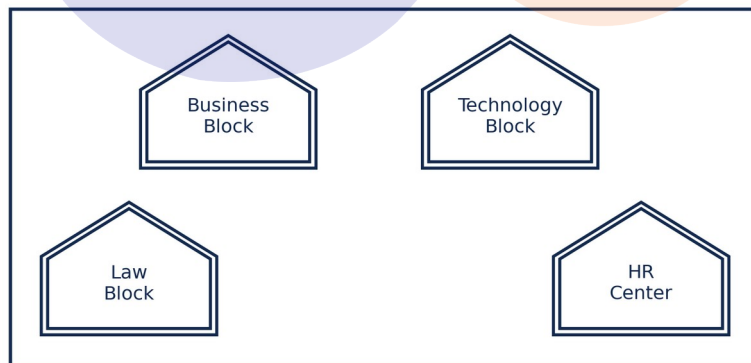
    if updated:
        print("Fees updated for Data Science students.")
    else:
        print("No student enrolled in Data Science found.")
```

*Note: import pickle may be treated as part of the complete program and need not be repeated in each part.*

**Marking Scheme** ½ mark for correctly defining the function header + ½ mark for correctly opening the file + 1 mark for using load ( ) with a while loop and a try ... except block + 1 mark for checking the condition and updating the value = **3 marks**.

**Question 37 — 5 Marks**

AKA University is setting up its Academic blocks at Raipur and planning to set up a network. The university has 3 academic blocks and one Human Resource Centre as shown in the diagram below:



**Centre-to-centre distances between various blocks:**

[ Distance table of the question paper to be inserted here ]

**Number of computers in each of the Blocks / Centre:**

[ Computer-count table of the question paper to be inserted here ]

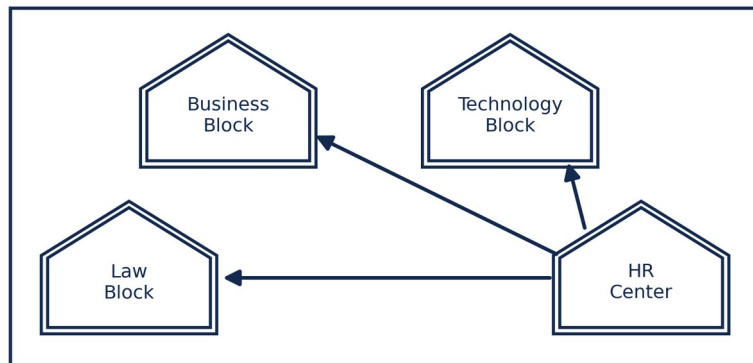
- I. Suggest the most suitable place (i.e. Block / Centre) to install the server of this university with a suitable reason.
- II. Identify and draw the most suitable topology for connecting all the blocks if the university wants a centralized network management system.
- III. Suggest the placement of the following devices with reasons: (a) Repeater (b) Switch
- IV. The Institute is planning to link its study centre situated in Delhi. Suggest a way to connect it with a reasonably high-speed guided medium.

V. (a) The Vice-Chancellor wants to hold a face-to-face digital meeting with the Heads of Departments in all blocks simultaneously. Which protocol will be used? **OR** (b) What is the use of VoIP?

#### Answer and Solution

**I. The HR Centre**, as it has the maximum number of computers. Placing the server where the largest number of nodes is located keeps most of the traffic local and minimises network delay.

**II. Star topology** — every block is connected directly to the HR Centre, which acts as the central node. This gives the centralized management the university wants, and a fault in one link does not affect the others.



**III. (a) Repeater** — to be placed on the HR Centre → Law Block link and on the HR Centre → Business Block link, because the distance on these links is more than 100 metres and the signal would otherwise become weak.

**(b) Switch** — one switch is to be placed in each block / centre, since a separate local network has to be formed inside every block and a switch connects those computers and forwards data only to the intended node.

**IV. Optical fibre cable** — it is a guided medium that offers very high bandwidth, very low attenuation over long distances and immunity from electromagnetic interference, so it is the best choice for linking the Delhi study centre.

**V. (a) VoIP (Voice over Internet Protocol).**

**(b) VoIP** is a technology that allows users to make voice calls and hold audio / video conversations over the Internet instead of a traditional telephone line, converting the voice signal into data packets. It is used for internet telephony, video conferencing and online meetings, and is far cheaper than conventional long-distance calling.

**Marking Scheme** 5 × 1 mark for each correct part = **5 marks**. For part V, either alternative (a) or (b) is to be attempted.