

**VIT**Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

SLOT

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
CONTINUOUS ASSESSMENT TEST - II
WINTER SEMESTER 2025-2026**

Programme Name & Branch : B.Tech CSE
Course Code and Course Name : BCSE207L – Programming for Data Science
Faculty Name(s) : Prof. M Rajasekhara Babu, Prof. NALINI N,
Prof.SARAVANAKUMAR K
Number(s) : VL2025260502059, 2541, 2543
Date of Examination : 18/03/2026
Duration : 90 minutes

Maximum Marks: 50

General instruction(s):

- **Answer All Questions.** M - Max mark; CO – Course Outcome; BL – Blooms Taxonomy Level (1 – Remember, 2 – Understand, 3 – Apply, 4 – Analyse, 5 – Evaluate, 6 – Create)
- **Course outcomes:** CO2. Design a suitable form for analysis from real-time data.
CO3. Formulate insights from the data through statistical inferences.

1. Given the following R program using the dplyr package, fill in the blanks with appropriate dplyr functions and briefly explain each one of them. [10 Marks][CO2][BL4]

```
library(dplyr)
students <- data.frame(
  StudentID = c(101,102,103,104,105,106,107,108),
  Name = c("Arun","Bala","Cathy","David","Eva","Farhan","Geetha","Hari"),
  Department = c("CSE","CSE","ECE","ECE","MECH","MECH","CSE","ECE"),
  Marks = c(85, 72, 58, 91, 66, 49, 77, 88)
)
students1 <- students %>% _____(StudentID, Name, Department, Marks)
students2 <- students1 %>% _____(Marks > 60)
students3 <- students2 %>% _____( Grade = case_when( Marks >= 85 ~ "A",
  Marks >= 70 ~ "B", Marks >= 60 ~ "C", TRUE ~ "D" ))
students4 <- students3 %>% _____(Department) %>%
  _____(Average_Marks = mean(Marks), Count = n())
print(students4)
```

2. Find and discuss the error or unnecessary instructions (if any) present in the following R script fragments. Also, write down the corrected versions. [Treat each of the following questions as individual questions.][Note: df, df1, cars are data frames] [10 Marks][CO2][BL3]

(a) The following code is intended to compute average marks per department:

```
df %>% group_by(dept) %>% mutate(avg = mean(score)) %>% summarise(avg_score = avg)
```

(b) df1 %>% inner_join(df2, by = "id") %>% group_by(department) %>% summarise(total = sum(salary))

(c) df %>% spread(key = subject, value = marks) %>% summarise(avg = mean(marks))

(d) The following code is intended to select cars whose price is above the average price of their dealer:
`cars %>% filter(price > mean(price)) %>% group_by(dealer_id)`

3. A dataset containing semester-wise marks of students is given below in wide format: [CO2][BL4]

Student	Sem1	Sem2	Sem3
A	78	81	85
B	65	70	72
C	88	90	93

(a) Explain why this dataset may not be suitable for performing grouped statistical analysis across semesters. (3 Marks)

(b) Describe how you would transform this dataset into a tidy (long) format using `gather()`. Clearly explain: (i) Which columns will be gathered, (ii) What the key column represents, (iii) What the value column represents. (4 Marks)

(c) After performing analysis, you need to present the results again in wide format. Explain how `spread()` can be used to reconstruct the original structure. (3 Marks)

4. An organization stores employee information in a **SQLite database** in a table named "Employee" in a database named "Company.db". Consider the following SQL queries: [10 Marks][CO3][BL3]

(a) `SELECT employee_id, name FROM employee WHERE gender = 'male' ORDER BY employee_id;`

(b) `SELECT gender, COUNT(*) FROM employee GROUP BY gender;`

Write an R script for each of the above queries individually that performs the following tasks:

1. Establish a connection to the SQLite database.
2. Execute the given SQL query directly from R and display the result.
3. Perform the same operation using functions from the `dplyr` package instead of directly writing the SQL query.
4. Properly close the database connection after retrieving the results.

5. A car dealership maintains its sales records in a JSON file. Each record contains the dealer name, city, and a list of cars sold by that dealer, where each car entry includes the car model and its price. Write an R script to perform the following tasks: [2+3+3+2 Marks][CO3][BL3]

1. Read the JSON file into R using an appropriate package such as `jsonlite`.
2. Convert the nested JSON structure into a data frame where each row represents one car sale with columns: dealer, city, model, and price.
3. Compute the total sales value for each dealer (using both aggregation function and `dplyr`).
4. Identify the dealer with the highest total sales value.
