	Semester- I	
	Paper - III	
Course Type: Core	Credit	Course Code: CS103
Title : Practical course on	Problem Solving using Comp	outer and 'C' programming
	and	
Ι	Database Management System	ns
Teaching Scheme	No. of Credits	Examination Scheme
3 Hrs / week	1.5	IE: 15 Marks
		UE: 35 Marks
Course Objectives		
• To understand the program d	levelopment life cycle.	
• Solve simple computational	problems using modular design	and basic features of the 'C'
language.		
• Understand basic database m	anagement operations.	
• Design E-R Model for given	requirements and convert the s	same into database tables.
Course Outcomes:-		
On completion of this course, stu	idents will be able to :	
<ul> <li>Devise pseudocodes and flow</li> </ul>	wchart for computational proble	ems.
• Write, debug and execute sir	nple programs in 'C'.	
• Create database tables in pos	tgreSQL.	
• Write and execute simple, ne	ested queries.	
Guidelines :		
Lab Book: The lab book is	to be used as a hands-on re	esource, reference and record of
assignment submission and co	mpletion by the student. The	he lab book contains the set of
assignments which the student n	nust complete as a part of this c	ourse.
Submission:		

Problem Solving Assignments:

The problem solving assignments are to be submitted by the student in the form of a journal containing individual assignment sheets. Each assignment includes the Assignment Title, Problem statement, Date of submission, Assessment date, Assessment grade and instructors sign.

Programming Assignments:

Programs should be done individually by the student in the respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

#### DBMS Assignments:

For each problem/case study, the student must design the database model in the form of an E-R

diagram. Table design should be based on the same and must include proper constraints and integrity checks. The students have to create, populate the tables and then perform the activities specified in each of the assignments. A pool of databases will get created as student progresses through the assignments and these databases can be repeatedly used in subsequent assignments. A separate softcopy of the queries must be maintained for each assignment.

#### Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes and good programming practices.

## **Operating Environment:**

For 'C' Programming : Operating system: Linux Editor: Any linux based editor like vi, gedit etc. Compiler : cc or gcc

## For DBMS:

Operating System: Linux Operating system DBMS: PostgreSQL Language: SQL

## Suggested List of Assignments:

## A) Problem Solving and C programming:

## Assignment 1.

Problem Solving using Pseudo code and Flowchart, Simple programs, Understanding errors and error handling.

# Assignment 2.

Decision Making Control Structures.

## Assignment 3.

Loop Control Structures

## Assignment 4.

Functions (User Defined functions, Library functions and Recursion ).

## Assignment 5.

Arrays (1-D and 2-D).

## **B)** Database Management Systems

### Assignment 1.

To create simple tables with only the primary key constraint ( as a table level constraint & as a field level constraint) (include all data types)

### Assignment 2.

To create more than one table, with referential integrity constraint, PK constraint. Assignment 3.

To create one or more tables with following constraints, in addition to the first two constraints (PK & FK)

- a. Check constraint
- b. Unique constraint
- c. Not null constraint

## Assignment 4.

To drop a table, alter schema of a table, insert / update / delete records using tables created in previous Assignments. ( use simple forms of insert / update / delete statements)

## Assignment 5.

To query the tables using simple form of select statement Select <field-list> from table [where <condition> order by <field list>] Select <field-list, aggregate functions > from table [where <condition> group by <> having <> order by <>]

## Assignment 6.

To query table, using set operations (union, intersect)

## Assignment 7.

To query tables using nested queries (use of 'Except', exists, not exists, all clauses **Assignment 8.** 

To create views.

Books: Laboratory handbook prepared by the University.

	Semester- II	
	Paper - III	
Course Type: Core	Credit	Course Code: CS203
Title : Practical Course of	n Advanced 'C' Programm	ing and Relational Dstabase
	Management Systems	
Teaching Scheme	No. of Credits	Examination Scheme
3 Hours / week	1.5	IE : 15 Marks
		UE: 35 Marks
Course Objectives		
• To solve real world computa	tional problems.	
• To perform operations on rel	ational database managemen	t systems.
Course Outcomes:-		-

On completion of this course, students will be able to :

- Write, debug and execute programs using advanced features in 'C'.
- To use SQL & PL/SQL.
- To perform advanced database operations. ٠

# **Guidelines :**

Lab Book: The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

# Submission:

# **Programming Assignments:**

Programs should be done individually by the student in respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

# **RDBMS** Assignments:

For each problem/case study, the student must design the database model in the form of an E-R diagram. Table design should be based on the same and must include proper constraints and integrity checks. The students have to create, populate the tables and then perform the activities specified in each of the assignments. A separate softcopy of the table creation statements and queries must be maintained for each assignment.

# Assessment

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes and good programming practices.

## **Operating Environment:**

For 'C' Programming : Operating system: Linux Editor: Any linux based editor like vi, gedit etc. Compiler : cc or gcc

For DBMS: Operating System: Linux Operating system DBMS: PostgreSQL 11 and higher Language: PL/SQL

#### Suggested List of Assignments:

# A) Advanced C Programming:

## Assignment 1.

Simple Pointers.

a) Pointer initialization and use of pointers.

b) Pointer Arithmetic.

#### Assignment 2.

Dynamic Memory Allocation.

### Assignment 3.

String handling using standard library functions.

#### Assignment 4.

Structure and Unions.

#### Assignment 5.

File Handling.

# Assignment 6.

C Preprocessors.

# **B)** Relational Database Management Systems:

## **Assignment 1: Stored Procedure**

- 1) A Simple Stored Procedure
- 2) A Stored Procedure with IN, OUT and IN/OUT parameter

## **Assignment 2: Stored Function**

- 1) A Simple Stored Function
- 2) A Stored Function that returns
- 3) A Stored Function recursive

## **Assignment 3 : Cursors**

- 1) A Simple Cursor
- 2) A Parameterize Cursor

## **Assignment 4 : Exception Handling**

- 1) Simple Exception- Raise Debug Level Messages
- 2) Simple Exception- Raise Notice Level Messages
- 3) Simple Exception- Raise Exception Level Messages

## Assignment 5 : Triggers

- 1) Before Triggers (insert, update, delete)
- 2) After Triggers (insert, update, delete)

Books: Laboratory handbook prepared by the University.