

**B.Sc. DEGREE PROGRAMME**  
**MATHEMATICS (ELECTIVE COURSE)**  
**MM6B01(E03) : C PROGRAMMING FOR**  
**MATHEMATICAL COMPUTING**

**3 hrs / week**

**2 credits**

**30 weightage**

**Course Requirements:**

Basic familiarity with computer.

A C compiler (Turbo C or Turbo C++) to do the Assignments in the course.

**Course Goals**

Students who complete this course will:

1. Have the basic skills required for computer programming.
2. Learn to write, compile and debug a C program
3. Be able to solve Mathematical problems using C programs
4. Introduce how to use C for drawing graphs and use mathematical models.

**Course Contents**

The course has Theory Part and Practical Part. The total periods for the course is 54 hrs of which 36 hrs for theory and 18 hrs for practical. Theory part focus on learning C Language to solve mathematical problems as listed in the Annexure 1. As and when various structures in C Language are discussed take examples from mathematics background as far as possible and conduct lab sessions to reinforce the idea. The Practical sessions are for testing the programs with the help of a C/C++ compiler. For entertaining free software, use Linux environment. A student must keep with the computer output obtained. He/She is expected to do a minimum of 12 programs selected from the list. (As shown in Annexure I) besides some other suitable sample programs to understand the style of C programming. The student has to maintain an observation note book and a practical record. Of the 3 periods per week 2 are for theory class and 1 is for practical session.

The University will conduct only theory examination, but Practical examination should be conducted internally and this should be considered for internal mark. For internal assessment minimum three tests, of which one should be practical test, may be conducted.

## **Theory Session (36 hrs)**

### **Text Books**

1. E. Balaguruswamy : Programming in Ansi C, Tata McGraw Hill.
2. Basavaraj S. Anami, Shanmughappa, A., Angadi S. & Sunilkumar S. Manvi – Computer Concepts and Programming (A Holistic Approach to Learning C), Prentice Hall of India.

### **Module I (9 hrs)**

Program Fundamentals: Computer Languages – Operating System – Compilation of Program – Different Types of Errors – Debugging of programs – Rewriting and Program Maintenance – Program Life Cycle (Text Book 2 – A quick review of Chapter 1 section 1.5).

Algorithms and Flow Charts: Algorithms and their characteristics – Flow Charts and their Uses – Advantages and Drawbacks of Flow Charts. (A quick view of Text Book 2 – Chapter 2 – All sections).

Overview of C: History of C – Importance of C – Sample programs – Basic Structure of a C Program – Programming Style – Executing C Program – DOS System (Text Book 1 – Chapter 1 – Section 1.1 to 1.12).

Constants, Variables and Data Types: Introduction – Character set – Keywords and Identifiers – Constants – Variables – Data Types – Declaration of storage class – Declaration of variables (Primary and User defined) – Assigning Values to variables – Symbolic constants (Text Book 1 – Chapter 2 – Sections 2.1 to 2.13).

### **Module II (9 hrs)**

Operators and Expressions: Introduction – Arithmetic operators – Relational operators – Logical operators – Increment and Decrement operators – Conditional Operators – Arithmetic Expression – Evaluation of Expressions – Precedence of Operators – Some Computational Problems – Type Conversions in Expressions – Mathematical Functions (Text Book 1 – Chapter 3 – Sections 3.1 to 3.7 and 3.10 to 3.16).

Managing Output Operations: Introduction – Reading and writing a character and a string – Formatted Input – Formatted Output – Use of %c, %d, %e, %f, %s – (Text Book 1 – Chapter 4 – Sections 4.1 to 4.5).

### **Module III (9 hrs)**

Decision Making and Branching: Introduction – If Statements (Simple if, if-else, nested if, ladder if) – Switch Statement – Conditional Operator – Go to Statement (Text Book 1 – Chapter 5 – Sections 5.1 to 5.9).

Decision Making and Looping: Introduction – While Statement – Do-while Statement – For Statement – Use of break, goto, continue in control statements (Text Book 1 – Chapter 6 – Sections 6.1 to 6.5).

### **Module IV (9 hrs)**

Arrays: Introduction – One Dimensional Arrays – Two Dimensional Arrays – Initialization of arrays – Multi Dimensional Arrays (Text Book 1 – Chapter 7 – Sections 7.1 to 7.9).

User Defined Functions: Introduction – Need for user defined functions – The form of C functions – Return of values – Calling a function – category of functions – Recursion – Function with arrays (Text Book 1 – Chapter 9 – Sections 9.1 to 9.10 and 9.16, 9.17, 9.18).

### **Reference Books**

1. K.R. Venugopal & Sudeer R. Prasad : Programming with C, Tata McGraw Hill.
2. Yashhant Kanetkar : Let us C, BPB Publication.
3. Byron Gottreid : Programming with C, Tata McGraw Hill.
4. V. Rajaraman : Computer Programming, Prentice Hall of India.

### **Lab Sessions (18 hrs)**

All the concepts in the theory sessions must be tested on a computer using a C compiler. A minimum of 12 problems from the list given in the Annexure I must be solved using C programming technique. A student should keep a practical record of the problem given by the teacher, algorithm, program and the output obtained in the lab session.

### **Practical Examination**

The practical examination of 1 hour duration is only for internal assessment.

### **Annexure I**

List of Numerical Problems

Section A (minimum 4)

1. Find GCD / LCM of two numbers.
2. Find the factorial of a number using recursion.
3. Check whether a number is prime or not
4. Reverse a n-digit number.
5. Find the sum of a set of numbers.
6. Write First n multiple of 7.
7. Find the maximum of two numbers using a function program.
8. Add two matrices / transpose a matrix.

Section B (minimum 5)

1. Find the maximum and the minimum of n numbers
2. Find the sum of the squares of first n natural numbers using loop
3. Find the number of above average student based on their mathematics marks

4. Multiply two matrices
5. Find the sum of digits of a n-digit number
6. Find the first n Fibonacci numbers
7. Evaluate Trigonometric / logarithmic / exponential function for a given x using its infinite series
8. Solve a given quadratic equation
9. Find the mean and standard deviation of a set of marks
10. Arrange a set of numbers in ascending / descending order.

Section C (minimum 3)

1. Integrate a function using trapezoidal rule
2. Solve a first order differential equation using Euler's method
3. Find a real root, if any, of polynomial equation using Bisection method.
4. Solve a first order differential equation using by Rung-Kutta method
5. Newton-Raphson's Method
6. Integrate a function using Simpson's rule.