

MCA – 12 Logic Development using Programming language

Examination Scheme					
Internal		External		Total	
Th. (Marks)	Pr. (Marks)	Th. (Marks)	Pr. (Marks)	Th. (Marks)	Pr. (Marks)
30	20	70	30	100	50

UNIT – I**(25%)****Overview of C :**

Introduction, Importance of C, Sample C programs, Basic structure of C programs, Programming style, Executive a C program.

Constants, Variables and data Types :

Introduction, Character Set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of Variables, Assigning values to variables, Defining symbolic constants.

Operators and Expression :

Introduction, Arithmetic of Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bit-wise Operators, Special Operators, Arithmetic Expressions, Evaluation of expressions, Precedence of arithmetic operators, Some computational problems, Type conversions in expressions, Operator precedence and associatively, Mathematical function.

Management Input and Output Operators :

Introduction, reading a character, writing a character, Formatted input, Formatted output.

UNIT – II**(25%)****Decision Making branching :**

Introduction, Decision making with IF statement, Simple IF statement, the IF ELSE statement, Nesting of IF ... ELSE statements, The ELSE IF ladder, The switch statement, the ternary (? :) Operator, the GOTO statement.

Decision Making Looping :

Introduction, the WHILE statement, the DO statement, The FOR statement, Jumps in loops Break and continue.

Array :

Introduction, One-dimensional arrays, Two-dimensional arrays, Initialization of two-dimensional arrays, Concept of Multidimensional arrays.

Handling of Character strings :

Introduction, Declaring and initializing string variables, Reading string from terminal, Writing string to screen, Arithmetic operations on characters, Putting string together, String Operations : String Copy, String Compare, String Concatenation And String Length, String Handling functions, Table of strings.

UNIT – III**(25%)****User-Defined Functions :**

Introduction, Need for user-defined functions, The form of C function, Return values and their types, Calling a function, category of functions, No arguments and no return values, Arguments with return values, Handling of non-integer functions, Nesting of functions, Recursion, Functions with arrays, The scope and Lifetime of variables in functions, ANSI C functions.

Structures and Unions :

Introduction, Structure definition, Giving values to members, Structure initialization, Comparison of structures, Arrays of structures, Arrays within structures, Structures within Structures, Structures and functions, Unions, Size of structures, Bit fields.

UNIT – IV**(25%)****Pointers :**

Introduction, Understanding pointers, Accessing the address of variable, Declaring and initializing pointers, Accessing a variable through its pointer, Pointer expressions, Pointer increments and scale factor, Pointers and arrays, Pointers and character strings, Pointers and Functions, Pointers and structures.

File Management in C :

Introduction, Defining files and its Operations, Error handling during I/O operations, Random access files, Command line arguments.

Dynamic Memory Allocation and Linked Lists :

Introduction, Dynamic memory allocation, Concepts of linked lists : Singly and Doubly linked list, Advantages of linked lists, Pointer revisited, Basic Operations On lists, Application of linked lists.

The Preprocessors :

Introduction, Macro Substitution, File inclusion, Compiler control directives, ANSI additions.

Developing a C program : Some guidelines :

Introduction, Program design, Program coding, Common programming errors, Program testing and debugging, Program efficiency.

Books :

ANSI C E. Balaguruswami

Let us C Kanitkar

Question Paper Scheme :**Section – I**

Q.1 Objective Type Unit I & II (11) Marks

Q.2 Unit – I **OR** Q.2 Unit – I (12) Marks

Q.3 Unit – II **OR** Q.3 Unit – II (12) Marks

Section – II

Q.4 Objective Type Unit III & IV (11) Marks

Q.5 Unit – III **OR** Q.2 Unit – III (12) Marks

Q.6 Unit – IV **OR** Q.3 Unit – IV (12) Marks

PRACTICAL LIST

1. Write a C program to display "Hello Computer" on the screen.
2. Write a C program to display Your Name, Address and City in different lines.
3. Write a C program to find the area of a circle using the formula.
 $\text{Area} = \text{PI} * r^2$
4. Write a C program to find the area and volume of sphere. Formulas are
 $\text{Area} = 4 * \text{PI} * R * R$ $\text{Volume} = 4/3 * \text{PI} * R * R * R$.
5. Write a C program to find the maximum from given three nos.
6. Write a C program to find that the accepted no is Negative, Positive or Zero.
7. Write a C program to convert centigrade into Fahrenheit.
Formula : $C = (F - 32) / 1.8$.
8. Write a C program to find the sum of digits of accepted no.
9. Write a C program to find the sum of first 100 natural nos.
10. Write a C program to find the sum of first 100 odd nos. and even nos.
11. Write a C program to display first 25 Fibonacci nos.
12. Write a C program to display first 100 prime nos.
13. Write a C program to find factorial of accepted nos.
14. Write a C program to print the accepted no and its reverse no.
15. Write a C program to find whether the accepted string number is palindrome or not.
16. Write a C program to find $x_1 + x_2 + x_3 + x_4 + \dots + x_n$.
17. Write a C program to find $1 + 1/2 + 1/3 + 1/4 + \dots + 1/n!$.
18. Write a C program to find $1 + 1/2! + 1/3! + \dots + 1/n!$.

19. Write a C program to convert decimal to binary.
20. Write a C program to convert decimal to octal.
21. Write a C program to convert decimal to hexa.
22. Write a C program to arrange the accepted numbers in ascending order and descending order.
23. Find log and square root of first 20 integer no. use arrays to store results.
24. Convert given line into upper case or lower case character as user want. Use switch statement for the choice of case.
25. Check accepted integer is prime number or no.
26. Convert accepted integer into word. For Example 55 = fifty five.
27. Convert accepted DATE into word. For Example 12/12/1972 = 12th December 1972.
28. Find the frequency of entered different integer nos.
29. Accept two different arrays, merge it and make it sort in ascending order.
30. Find smallest character from each word of accepted line.
31. Find longest line from given text.
32. Count How many Characters, Words, lines, spaces, tabs into given text.
33. Find the median from given nos.
34. Sort the accepted string in ascending order.
35. Print 3 students detail of R-no, name, address, city, phone on screen. Use structure.
36. Simple pointer practical for printing integer and its memory address.
37. Print the multiply vale of two accepted numbers.
38. Find the NPR, NCR with using User Defined Function.
 ${}^N P_R = N!/(N-R)!$ ${}^N C_R = N!/(R!(N-R))$.
39. Swap the values of two different numbers using UDF. Use pointer and function ...
40. Create one text file store some information into it and print the same information on terminal.
41. Create one integer value contained file. From this file create another two files one for odd and second for even numbers print the result of both the files.
42. Create one file insert some information into it. Using fprintf() and fscanf() function.
43. Find the percentage of vowels in accepted text. Use file ...
44. Display this kind of output on screen.

C

CP

CRR

.

CPROGRAMING

.

CPR

CP

C

45. Display this kind of output on screen.

1

01

101

0101

...

46. Display this kind of output on screen.

1

01

101

...

47. Display this kind of output on screen.

1

10

101

...

48. Display this kind of output on screen.

1

10

101

...

49. Display this kind of output on screen (in the left of the screen)

1

22

333

4444

50. Display this kind of output on screen (in the center of the screen.)

1

22

333

4444

...

51. Display this kind of output on screen.

1

23

456

78910

...

..... 90 91

52. Display this kind of output on screen.

1

23

456

78910

53. Display this kind of output on screen.

*

**

54. Display this kind of output on screen.

*

**

55. Display this kind of output on screen.

0000000

000000

000

**

0

56. Display this kind of output on screen.

1

121

12321

1234321

123454321

57. Display this kind of output on screen.

```
1
232
34543
4567654
567898765
```

58. Display this kind of output on screen.

```
1
23
345
4567
56789
...
..... 90 91
```

59. Write a program to work as a dos type command using command line argument.

60. Write a program to work as a dos copy command using command line argument.