MAEER' Arts, Commerce and Science College

F.Y. B. Sc. (Computer Science)

Subject: Introduction to Programming & 'C' Programming

Question Bank Chapter wise (Theoretical)

Chapter 1: Programming Languages

1. What are different types of computer languages?
2. Explain machine language.
3. Distinguish between Assembly language & Higher level language.
4. What is the difference between syntax errors & logical errors?
5. What are compilers? How do they differ from interpreters?
6. What is an Assembler?
7. What are source programs & object programs?
8. Compare and Contrast Interpreter & Compiler.
9. State advantages of high level & assembly language.
10. State two interpreters & two compilers.

Chapter 2: Problem Solving using Computers

1. What is an algorithm? What are the advantages of writing an algorithm?
2. What care should be taken while writing an algorithm?
3. Write an algorithm to calculate prime factors of an integer.
4. What is a flowchart? What are the principles of flowcharting?
5. Explain the flowcharting symbols with examples?
6. Draw a flowchart to calculate the sum & average of n numbers.
7. Draw a flowchart to check whether a given number is prime or not.
8. Draw a flowchart to generate N terms of Fibonacci series.
9. Draw a flowchart to calculate $\mathbf{x^n}$ where x is real and n is an integer.
10. Draw a flowchart to find all divisors of an integer number N.
11. Accept the day of the week and display whether the day is a 'Working Day' or 'Week End'. Display proper 'Error Message' for all incorrect inputs.
12. Draw a flowchart to print the pattern:

    1

    1 2

    1 2 3

    1 2 3 4

    1 2 3 4 5

Chapter 3: Introduction to C

1. Where was C developed by whom?
2. State the features of C language.
3. State the applications of C language.
4. Explain structure of C program.
5. What is the purpose of main ( ) function?
6. Why C called middle – level language?
7. Explain program development life-cycle.
8. What is the purpose of compiling?

Chapter 4: C tokens

1. State basic data types in C.
2. State user-define data types in C.
3. Explain the use of sizeof () operator.
4. Discuss the logical operators of C.
5. Discuss the relational operators of C.
6. Discuss the bitwise operators of C.
7. Define type casting with examples.
8. Explain working of conditional operator.
9. What is an escape sequence?
10. State the use of increment and decrement operators.
11. State the use of bitwise operators.
12. State the use of shift operators.
13. How the symbolic constants are define.
14. What is purpose of #include?

Chapter 5: Input and Output

1. State any four functions included in the header file ctype.h.
2. Which header file is use for input-output operations?
3. State two functions for reading the characteristics from the user.
4. What is the difference between puts( ) and putc( )
5. What is the purpose of scanf( ) function? State any four format specifies use with scanf( ).
6. What is the purpose of printf( ) function? State any four format specifiers use with printf( ) statement with an example.

Chapter 6: Control Structures

1. What do you mean by sequential execution of a program?
2. What do you mean by repetition?
3. What is compound statement? Explain with the help of syntax diagram.
4. What is meant by control structure?
5. What is purpose of the while structure? Explain the execution of the while structure.
6. What is the purpose of do-while structure? Differentiate between while a do-while structures.
7. Explain the for control structures. Also, explain the role of control variables.
8. Write a short note on the nesting of control structures.
9. Draw a neat syntax diagram of the if statement and explain its execution with the help of suitable example.
10. Explain the purpose of switch statement. How does this structure differ from the others? Compare switch statement with if-else statement.
11. What is meant by label? What is the use of label?
12. What is a purpose of goto statement?
13. What are the drawbacks of using goto statement in a structured programming language?
14. What is a use of control structure?
15. Why compound statements are useful?
16. What is looping?
17. Differentiate between conditional and unconditional branching.
18. What is branching?

Chapter 7: Functions in C

1. What are the advantages of program modularization? How is programs modularization achieved in C?
2. What is meant by scope of variables?
3. Distinguish between local and global variables.
4. What is meant by parameters passing? How are they used?
5. What is difference between actual parameters and formal parameters?
6. What is a function? How are the functions declared?
7. Explain the concept of recursive with the help of suitable example.
8. Write recursive function to convert decimal number into binary number string.
9. Write a note on storage class of C.
10. What is the difference between automatic and static storage class.
11. Write a function to add all even numbers together and add all odd numbers together and print the even sum and odd sum. Accept the number range from the user.
12. Write a function to find out the GCD of 2 given integers.
13. Write a function to check whether the given year is a leap year or not. If it is leap year print 'Yes' else print 'No'.
14. Explain how to return more than one values from a function.

Chapter 8: Arrays

1. What is meant by array? Explain with the help of examples.
2. What do you mean by an index or subscript? What data types can be used for an index?
3. How do you declare an array? How do you access an individual array element?
4. What is meant by dimensions of an array? Explain two-dimensional array with suitable example. How do you declare two-dimensional array?
5. Why nested loops are required in some applications of multidimensional arrays?

Chapter 9: Pointers

1. What is a pointer? How is pointer initialized?
2. Differentiate between call by value and call by reference parameter passing techniques, with the help of examples.
3. Which are the advantages of dynamic memory allocation over static memory allocation?
4. Explain pointer arithmetic with example.
5. Illustrate the use of pointer to function by example.
6. Distinguish between (*a) [5] and *a [5].
7. Write a function to interchange the content of two variables (use pointer parameters)
8. Write a function which takes pointer to integer array as an argument and find the smallest number from array. Implement this function in main ( ).
9. Write a function using pointers to add two matrices and to return the resultant matrix to the calling function.
10. Given the following declarations

    int a = 5, b=10;

    int *x = &a, *y=&b;

    Find the value of following expressions?

    (i) (*x)++

    (ii) – (*x)

    (iii) *x + (*y) --

    (iv) ++ (*y) - *x

Chapter 10: Strings

1. What are strings? How are they initialized?
2. What is the difference between scanf (“%s”) and gets ( ) function?
3. What do you mean by pointer to string? Explain with example.
4. Write a program that reads a string from a keyboard and check whether it is a palindrome or not.
5. Write a program to rewrite a given string in alphabetical order.
6. Example: The string NIRALI should be written as AIILNR.

7. Write a program to read list of cities from a keyboard and store into array. Search the given city in the list.
8. Write a program to read a string of alphabets, copy uppercase letters in 'upper' array and lowercase letters in 'lower' array.
9. Write a function which takes a string as an argument and returns the length of it without using standard library functions strlen ( ).

Chapter 11: Structures and Unions

1. How does the structure differs from array?
2. How does the union differs from structure?
3. Write note on nested structures.
4. What is the use of structure? Explain with example.
5. How is a structure declared and initialized? Give an example.
6. Can a function return a value of type 'pointer to structure?
7. What is the need for array of pointers to structures? Explain with an example.

Chapter 12: C preprocessor

1. What is command line argument?
2. Define command line argument? Specify the two arguments in detail.
3. Draw a data structure design of argc and argv where we have seven string pointers.
4. How to access command line argument? Explain with the help of simple program.
5. Write a program to copy the contents of one text file to other text file using command line argument.

Chapter 13: File Handling

1. Write a program that appends on file at end of another.
2. Write a program to copy the content of one file into another using command line arguments.
3. Write a program to count number of occurrences of all alphabets a to z from a given text file.

F.Y. B. Sc. (Computer Science)

Subject: Introduction to Programming & 'C' Programming

Question Bank (Programming)

1. WAP to convert upper case character to lower case using function
2. WAP to find largest / maximum among three numbers
3. Swap two number (using 3^{rd} variable or without 3^{rd} variable)
4. To check whether the character is vowel or not using function
5. To find factorial of a number & Power of a number
    i. By simple way
    ii. By using user defined function
    iii. Recursion function
6. To find sum of digit of a number. eg. 3582=3+5+8+2=18
7. To check whether entered year is leap year or not
8. To print equivalent ASCII value of digits from 1 to 255
9. To print Armstrong number between 1 to 500
10. To check the number is prime or not
11. To check the number is palindrome or not
12. To print n terms of Fibonacci series
13. To check the given number is perfect or not
14. WAP menu driven for Arithmetic operation
15. WAP for multiplication & division without using * & / operator
16. WAP to calculate sum of following series

    $x \quad x^2 \quad x^3 \quad \ldots \quad x^n$

i) $1+\frac{}{1!}+\frac{}{2!}+\frac{}{3!}+\frac{}{n!}+\ldots\ldots$

ii) $1+\frac{x}{1!}+\frac{x^2}{2!}-\frac{x^3}{3!}+\frac{x^4}{4!}-\frac{x^n}{n!}+\ldots\ldots$

iii) $1-\frac{1}{2^2}+\frac{1}{3^2}-\frac{1}{4^2}+\frac{1}{n^2}-\ldots\ldots$

iv) $\frac{1}{(x+1)}+\frac{2}{(x+1)^2}+\frac{3}{(x+1)^3}+\frac{4}{(x+1)^4}+\ldots\ldots\ldots\frac{n}{(x+1)^n}$

v) $\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\ldots\ldots\ldots\ldots\frac{1}{n}$

vi) $\frac{(x-1)}{x}+\frac{1}{2}\frac{(x-1)^2}{x^2}+\frac{1}{3}\frac{(x-1)^3}{x^3}$

vii) $\sin x = x-\frac{x^3}{3}+\frac{x^5}{5}-\frac{x^7}{7}+\ldots\ldots$

1! 2! 3!

viii) $1^2-2^2+3^2-4^2$ ..... n^2

17.WAP to print following patterns

1)
```
* * * *
* * * *
* * * *
* * * *
```

2)
```
1 2 3 4
1 2 3 4
1 2 3 4
1 2 3 4
```

3)
```
1 1 1 1
2 2 2 2
3 3 3 3
4 4 4 4
```

4)
```
2 4 6 8
2 4 6 8
2 4 6 8
```

5)
```
1 2 3
4 5 6
7 8 9
```

6)
```
A B C D
A B C D
A B C D
A B C D
```

7)
```
A A A A
B B B B
C C C C
D D D D
```

8)
```
A B C
D E F
G H I
```

9)
```
*
* *
* * *
* * * *
```

10)
```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

11)
```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

12)
```
A
A B
A B C
A B C D
A B C D E
```

13)
```
A
B B
C C C
D D D D
```

14)
```
1
2 3
4 5 6
7 8 9 10
```

15)
```
2
2 4
2 4 6
2 4 6 8
```

16)
```
1
2 2
3 3 3
4 4 4 4
```

17)
```
5 5 5 5 5
4 4 4 4
3 3 3
2 2
```

5 5 5 5 5 1

18)
1
1 2
1 2 3
1 2 3 4
1 2 3
1 2
1

19)
1*
1* 2*
1* 2* 3*
1* 2* 3* 4*

20)
*
* * *
* * * * *
* * * * * * *

21)
1
2 1 2
3 2 1 2 3
4 3 2 1 2 3 4

22)
1 2 3 4 3 2 1
1 2 3 4 3 2 1
3 4 3
4

23)
0 1 2 3 4 5 4 3 2 1 0
0 1 2 3 4 3 2 1 0
0 1 2 3 2 1 0
0 1 2 1 0
0 1 0
0

18.WAP to swap a two number using user defined function

19.WAP function for Fibonacci series

20.WAP for add, sub, multi, div using switch & user defined function

21.WAP to sort array in ascending order

22.WAP to search a element in an array (linear search)

23. To print odd & even elements of array separately
24. For addition of two array
25. Print array in reverse order
26. Find average of an array elements
27. Enters today's date & find tomorrows date
28. Multiplication & subtraction of 1D array
29. To print min element & max element of an array
30. Transpose of given matrix
31. Addition of 2 matrix
32. Print right diagonal, left diagonal, upper triangular, lower triangular of matrix using switch
33. Matrix multiplication
34. To check the given matrix is identically matrix or not
35. To find sum of rows & sum of columns of matrix
36. Calculate string length
37. Copy one string to another without using library function
38. Concatenation of two string without using library function
39. Comparison of two string whether they are equal or not
40. To find maximum of two string
41. Reverse a string
42. To print rotation of a string (eg. WEL ELW LWE)
43. Input 5 names & display & find length of each name
44. WAP to print string in following format if str="WEL"

i.
```
W
W E
W E L
```

ii
```
W
W E
W E L
```

iii
```
W
W E W
W E L E W
```

Using Pointers the following program should be performed

1. WAP for addition of two numbers using pointers
2. WAP for swapping of two numbers using pointers & function
3. WAP to input 10 values & display using pointers
4. WAP to print array in reverse order (1 D)
5. Addition of two array (2 D) using function
6. WAP to find length of string
7. WAP to accept two strings & compare them using pointer
8. WAP that copies one string into another string
9. WAP for concatenation of two string
10. WAP to print given number in reverse order
11. WAP to check given number is Armstrong or not
12. WAP to find sum of digit
13. Multiplication of two number without using multiplication operator
14. Division of two number without using division operator
15. WAP to check given number is prime or not
16. WAP to check given number is perfect or not
17. WAP to print 2 D array
18. To find recursive sum of digit 88=8+8=16=1+6=7
19. WAP to find factorial of number
20. WAP to find power of number
21. WAP to print Fibonacci series of a n numbers

Structure

1. Create a structure student (roll number, name, marks of 3 subjects, percentage).

Accept details of n students and write a menu driven program to perform the following

operations. Write a separate function for the different options.

i] Search

ii] Modify

iii] Display all students' details

iv] Display student having maximum percentage

2. Create a structure empolyee (id, name, salary).Accept details of n employee and write a

menu driven program to perform the following operations. Write a separate function for

the different options.

i] Search by name

ii] Search by name

iii] Display all employees' details

iv] Display all employees having salary>_____.

Command Line Argument & Preprocessor Directive

1. Write a program to accept three integers as command line arguments and find

minimum, maximum and average of three. Display error message if invalid nmber of

arguments are entered.

2. Write a program which accept two string and two character as command line
   arguments and replace all occurrences of the first character by the second.
3. Define a macro EQUALINT which compares two parameters x and y and give 1 if
   equal and 0 otherwise. Use this macro to accept pairs of integers from the user.
   Calculate the sum of both.
4. Define a macro EQUALSTR which compares two strings x and y and give 1 if
   equal and 0 otherwise. Use this macro to accept two strings from the user and check if
   they are equal.