VALLIAMMAI ENGINEERING COLLEGE

SRM NAGAR, KATTANGULATHUR – 603 203

FIRST SEMESTER B.E / B.Tech., (Common to all Branches)

QUESTION BANK GE 6151 – COMPUTER PROGRAMMING

UNIT I INTRODUCTION

Generation and Classification of Computers- Basic Organization of a Computer –Number System – Binary – Decimal – Conversion – Problems. Need for logical analysis and thinking – Algorithm – Pseudo code – Flow Chart.

2 MARKS

1. Why computer is called an idiotic genius?
2. Differentiate between analog and digital computers.
3. State the characteristics of computers.
4. How will you classify computer systems?
5. What are the different components of a computer?
6. What are the advantages and disadvantages of using the first generation computers?
7. Which technology was used in the second generation computer and how was it better than the technology used in the first generation computers?
8. Give the advantages and disadvantages of third generation computers.
9. Convert the binary number **100110** into its octal equivalent.
10. Determine the decimal equivalent of the hexadecimal number **AC.C8.**
11. Convert **0.4375** decimal to binary system.
12. Convert the binary number **11000110** into Hexadecimal number.
13. Differentiate between RAM and ROM.
14. Draw a flowchart to find the maximum among the three numbers.
15. Compare and contrast flowchart and algorithm.
16. What is meant by pseudo code?
17. What is an algorithm?
18. Write an algorithm to compute the factorial of a number.
19. Write the pseudo code to find the given year is a leap year or not.
20. Give the advantages and limitations of Pseudo code.

16 MARKS

1. (i) Describe the characteristics of the computer.
   (ii) Explain briefly the developments in computer technology starting from a simple calculating machine to the first computer.
2. Explain in detail the different generation of computers.
3. Describe the different classification of computers.
4. Explain in detail about the various components of a computer system with block diagram. (or) Explain the organization of a computer.
5. Explain the various types of computer memory.
6. Convert the following:
   (i) Convert $(6245.14)_8$ to its decimal equivalent.
   (ii) Convert(111001.101)2 to its decimal equivalent.
   (iii) Convert the following numbers into their binary equivalent.
   a. $(59.6825)_{10}$ b. $(EBC)_{16}$ c. $(654)_8$
   (iv) Convert the following numbers into their binary equivalent.
   a. FAC_{16} b. 561_8
7. Explain the program development life cycle in detail.
8. Explain the need for an algorithm and highlight its advantages. Write an algorithm to find the greatest among three numbers.
9. Mention the guidelines in detail while drawing a flowchart with examples and list out the merits and demerits of flowcharting.
10. Explain pseudo code with an example and briefly discuss the different pseudo code structures. Differentiate algorithm, flowchart and pseudo code.

QUESTION BANK GE 6151 – COMPUTER PROGRAMMING

UNIT II C PROGRAMMING BASICS

Problem formulation – Problem Solving - Introduction to ' C' programming – fundamentals structure of a 'C' program – compilation and linking processes – Constants, Variables – Data Types – Expressions using operators in 'C' – Managing Input and Output operations – Decision Making and Branching – Looping statements – solving simple scientific and statistical Problems.

2 MARKS

1. Why header files are included in 'C' programming?
2. What is a global variable?
3. List the different data types available in 'C'?
4. What are keywords?
5. What do you mean by variables in 'C'?
6. What is ternary operator or conditional operator?
7. What is the difference between Logical AND and Bitwise AND?
8. What is the difference between '=' and '==' operator?
9. What is the use of sizeof( ) operator?
10. What are the escape sequences present in 'C'
11. What is the output of the programs given below? State the reason.

```
main()
{
float a;
int x=6, y=4;
a=x/y;
printf( "Value of a=%f" , a);
}
```

```
main()
{
float a;
int x=6, y=4;
a=(float) x/y;
printf( "Value of a=%f" ,a);
}
```

12. What are the types of I/O statements available in 'C'?
13. What is the significance of control string in Input/Output Statements?
14. Why we don't use the '&' symbol while reading a string through scanf() function?
15. What is the difference between ++a and a++?
16. Compare switch( ) and nested-if statement.
17. What is the difference between while loop and do…while loop?
18. Construct an infinite loop using while?
19. What is the difference between while(a) and while(!a)?
20. Differentiate break and continue statement

16 MARKS

1. Explain in detail the structure of a C program with an example.
2. Explain in detail about the constants, expressions and statements in 'C'.
3. Discuss about the various data types in 'C'.
4. Describe the various types of operators in 'C' language along with its priority.
5. Explain about the various decision making and branching statements.

6. Write short notes on the following:
   a. ‘for’ loop
   b. ‘while’ loop
   c. ‘do..while’ loop
7. Explain briefly about the input and output functions in ‘C’.
8. (a) Describe in detail about type conversions in ‘C’ with example.
   (b) Define delimiters. List them. Give an example program using various delimiters.
9. Explain the following:
   i. Keywords ii. Identifiers
   iii. C character set iv. Constants and volatile variables.
10. Write a program for the following :
   a. To check whether a given year is leap or not.
   b. To find the roots of a quadratic equation.
   c. To find the area and circumference of a circle with radius r.
   d. To convert the temperature given in Fahrenheit to Celsius.
   e. To calculate simple interest and the maturity amount.
   f. To find area of a triangle whose sides are a, b and c.
   g. To find the sum of first 100 integers.
   h. To find the sum of all odd / even numbers between 1 and 100.
   i. To check whether a number is prime or not.
   j. To find the digits of a number. (123 => 1+2+3=6)
   k. To reverse the digits of a number. (123 => 321)
   l. To check whether a given number is a palindrome or not. (232)
   m. To check whether a given number is perfect. (6=>1+2+3, 28=>1+2+4+7+14)
   n. To print the integers between 1 and n which are divisible by 7.
   o. To generate the first n numbers in a Fibonacci series.
   p. To find the factorial of a given number.
   q. To generate Armstrong number between 100 and 999.
   r. To find the average of n numbers.
   s. To find the sum of series :
      i. 1+(1+2)+(1+2+3)+(1+2+3+4)……n terms
      ii. $1^2+2^2+3^2+4^2$+……..n terms
      iii. 1+1/2+1/3+1/4+…..n terms
      iv. $Sin(x) = x - x^3/3! + x^5/5! - x^7/7! + \ldots + x^n/n!$
      v. $Cox(x) = 1 - x^2/2! + x^4/4! - x^6/6! + \ldots + x^n/n!$
   t. To print the following :
      i. Pyramid of digits

```
          1
        2 3 2
      3 4 5 4 3
    4 5 6 7 6 5 4
    ……................
```

      ii. Floyd’s triangle

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

UNIT III ARRAYS AND STRINGS

Arrays – Initialization – Declaration – One dimensional and Two dimensional arrays. String – String operations – String Arrays. Simple programs – sorting – searching – matrix operations.

PART – A (2 MARKS)

1. Define an Array. Give example.
2. List out the features of Arrays.
3. Is it possible to declare an array subscript with float data type?
4. What are the main elements of an array declaration?
5. What are the drawbacks of Initialization of arrays in C?
6. What will happen when you access the array more than its dimension?
7. What are the different ways of initializing array?
8. How to create a two dimensional array?
9. What is the use of '\0' and '%s'?
10. What is a String?
11. What is the starting index of an array?
12. Is address operator used in scanf() statement to read an array? Why?
13. What is the role of strrev()?
14. Distinguish between one dimensional and two dimensional arrays.
15. How to initialize a string? Give an example.
16. Differentiate between Linear search and Binary search.
17. Write the output of the following Code:

```
main()
{
   char x;
   x = 'a';
   printf("%d \n",x);
}
```

18. Specify any two methods of sorting.
19. List out the operations that are performed on character strings.
20. Write the output of the following Code:

```
main()
{
   static char name[]="Kagz  WrxAd"
   int  i=0;
   while(name[i]!='\0')
   {
     printf("%c",name[i]);
     i++;
   }
}
```

PART – B (16 MARKS)

1(i) Explain the need for array variables. Describe the following with respect to arrays: Declaration of array and accessing an array element. (8)

(ii) Write a C program to re-order a one-dimensional array of numbers in descending order. (8)

2. Explain the following functions with examples. (4+4+4+4=16)

(i) strlen() (ii) strcpy() (iii)strcat() (iv) strcmp( )

3. Write a program in C to find whether the given string is palindrome or not without using string functions. (16)

4. Write a C program to count the number of characters, spaces, vowels, consonants and others with using string functions. (16)

5. Describe the following with suitable examples. (8+8=16)
   (i) Initializing a 2 Dimensional Array
   (ii) Memory map of a Dimensional Array

6. Explain about the String Arrays and its manipulation in detail. (16)

7. (i). Write a C program to find average marks obtained by a class of 30 students in a test. (10)
   (ii).Write short notes on Reading and Writing string. (6)

8. Write a C program to merge two sorted array into a single sorted array. (16)

9. Write a C program to search an element from the array. (16)

10. Write a C program to perform the following matrix operations: (16)
    (i) addition (ii) subtraction (iii) multiplication (iv) transpose

UNIT IV FUNCTIONS AND POINTERS

Function – definition of function – Declaration of function – Pass by value – Pass by reference – Recursion – Pointers – Definition – Initialization – Pointers arithmetic – Pointers and arrays – Example Problems.

PART – A (2 MARKS)

1 What is pointer? How will you declare it?
2 What is a pointer to a pointer?
3 What are the operations that can be performed on pointers?
4 What is pointer arithmetic?
5 What is a void pointer and a null pointer?
6 What are formal parameters and actual arguments
7 Why is pointer arithmetic not applicable on void pointers?
8 What is user-defined function?
9 What is meant by library function?
10 Write the syntax for function declaration
11 What are the two parts of function definition?
12 Write the general form of header of a function
13 What is meant by pass by value and pass by reference?
14 What is a function call? Give an example of a function call
15 What is default arguments and command line arguments?
16 What is a recursive function?
17 Differentiate Direct and Indirect recursion
18 Differentiate Tail and Non Tail recursion
19 What is linear recursion?
20 What is a function pointer?

PART – B (16 MARKS)

1 Discuss about pointers and their operations that can be performed on it
2 What is an array of pointers and what is pointer to an array? Explain in detail with example.
3 Write in detail about function declaration and function definition
4 Discuss about the classification of functions depending upon their inputs and output (parameters)
5 Explain in detail about Pass by Value and Pass by reference.
6 Discuss about passing arrays to function.
7 Explain in detail about recursive function with sample code.
8 Explain in detail about function pointers.
9 Write notes on fixed argument functions and variable argument functions.
10 What are the applications of recursive function? Explain about Tower’s of Hanoi Problem.

UNIT V STRUCTURES AND UNIONS

Introduction – need for structure data type – structure definition – Structure declaration – Structure within a structure - Union – Programs using structures and Unions – Storage classes, Pre-processor directives.

PART – A (2 MARKS)

1 What is structure? Write the syntax for structure.
2 Write the various operations on structure.
3 How the members of structure object is accessed?
4 Write the use of size operator on structure.
5 What is a nested structure?
6 How typedef is used in structure?
7 Define Union in C.
8 Write the operations on union object.
9 Write the storage classes in C.
10 What is C preprocessor?
11 What is Translator?
12 What is compiler?
13 What is trigraph replacement?
14 What is line splicing?
15 What is tokenization?
16 What is macro? What are predefined macros?
17 What is pre processor directive handling?
18 What is line directive?
19 What is error directive?
20 Define conditional compilation directive.

PART – B (16 MARKS)

1 Explain about storage class specifiers.
2 Explain about C pre processor and phases of translation.
3 Explain about pre processor directive.
4 Explain about union and its practical applications.
5 What is enumeration explain with examples.
6 Explain functions and structures.
7 Explain about structures and its operations.
8 Explain about pointers to structures, array of structures and nested structures.
9 Write a C program using structures to prepare the students mark statement.
10 Write a C program using unions to prepare the employee pay roll of a company.