**B.E/B.Tech DEGREE EXAMNIATION ,NOVEMBER/DECEMBER 2015**

**FIRST SEMESTER**

**GE6151 : COMPUTER PROGRAMMING**

**(Common to all branches)**

**(Regulation : 2013)**

**Time : Three Hours Maximum :100 Marks**

**Answer ALL question**

**PART – A (10 X 2 = 20 Marks)**

1. **What is pseudo code?**

Pseudo code is made up of two words: Pseudo and code. It means imitation and code refers to instructions, written in a programming language. Pseudo code is an outline of a program. It uses plain English statements rather than symbols. It is also knows as PDL (Program Design Language).

1. **What is an algorithm?**

Algorithms are one the most basic tools that are used to develop the problem solving logic. An algorithm is defined as a finite sequence of explicit instruction that when provided with a set of input values process an output and then terminates. Algorithms can have steps that repeat or require decisions until the task is completed.

1. **What is compilation process?**

During the compilation process, the source program instructions are translated into a form that is suitable for execution by the computer. The translation process checks each and every instruction for correctness and if no errors are reported then it generates the object code.

1. **Discuss the working of modulo operator.**

Modulo operator is used to return the remainder.

1. **Declare a character array of size 5 and assign vowels to it.**

char   vowel[5] = {'a', 'e', 'i', 'o', 'u', '\0'};

1. **Give some examples of string functions.**

Strlen(),strcpy(),strcmp(),strcat(),strchr().

1. **What is function definition?**

A function definition is an independent program that performs a specific task. Function definition includes function header and function body. Function body contains list of instructions enclosed within a pair of brackets.

The general syntax of function definition:

<return-type> fun-name(argument/parameter list)

{

// body of the function

}

1. **What is an address operator and indirection operator?**

Indirection operator:  
The indirection operator (**\***) accesses a value indirectly, through a pointer. The operand must be a pointer value. The result of the operation is the value addressed by the operand; that is, the value at the address to which its operand points. The type of the result is the type that the operand addresses

Address operator:

The address-of operator (**&**) gives the address of its operand. The operand of the address-of operator can be either a function designator or an l-value that designates an object that is not a bit field and is not declared with the **register** storage-class specifier.

1. **Write a note on register storage class.**

Registers are able to store only restricted size value, so most of the compilers will allow only int and char variables to be placed in the register. Registers are limited in count, therefore it is important to select required variables for this purpose. The C compiler automatically converts **register** variables into non register variables once the limit is reached,

X.

**Syntax:**

register int count;

1. **What is the use of #define pre-processor?**

#define this macro defines constant value and can be any of the basic data types.

**PART-B ( 5 X 16 =80 )**

1. a) Explain in detail with block diagram about the digital computer organisation and discuss the function of each block. **16**

**Ref section 2.1**

**OR**

b) Perform the following: **(4 X 4 =16)**

(i) (1011.11011)**2  = ( )10**

**23 22 21 20. 2-1 2-2 2-3 2-4 2-5**

1 0 1 1 . 1 1 0 1 1

**8+0+2 +1 . 0.5+0.25+0+0.0625+0.03125**

**( 11.84375 )10**

(ii) (10111)2 x (1011)2  = ?

1 0 1 1 1

(X)

1 0 1 1

1 1 1 1 1 1 0 1

(iii) (D8BC)H = ( ? )2

(1101 1000 1011 1100)2

(iv) (4871)10 = ( ? )8

8 4871

8 608 7

8 76 0

8 9 4

8 1 1

1. a) What are the various operators available in c? Discuss each one of them with suitable examples?

**Ref section 8**

**OR**

b) Explain in detail about various decision making structures available in C with illustrative examples. **Refer 10**

1. a) Write a C program for finding the largest element and smallest element in a matrix.

#include <stdio.h>

int main()

{

    int arr[100];

    int i, max, min, size;

printf("Enter size of the array: ");

    scanf("%d", &size);

    printf("Enter elements in the array: ");

    for(i=0; i<size; i++)

    {

        scanf("%d", &arr[i]);

    }

    max = arr[0];

    min = arr[0];

    for(i=1; i<size; i++)

    {

              if(arr[i]>max)

        {

            max = arr[i];

        }

         if(arr[i]<min)

        {

            min = arr[i];

        }

    }

    printf("Maximum element = %d\n", max);

    printf("Minimum element = %d", min);

    return 0;

}

**OR**

b) Write a C program to multiply two matrices. **Refer 12.9.2**

1. a) Discuss about call by value and call by reference with illustartions. **Refer 14.7**

**OR**

b) What is recursion ? Explain a recursive function with suitable example. Write a

recursive function to find the factorial of a number. **Refer 14.9**

#include<stdio.h>  
int fact(int);

void main(){

  int num,f;

  printf("\nEnter a number: ");

  scanf("%d",&num);

  f=fact(num);

  printf("\nFactorial of %d is: %d",num,f);

  }

int fact(int n){

   if(n==1)

       return 1;

   else

       return(n\*fact(n-1));

 }

1. a) What is a structure? Create a structure with data members of various types and declare two structure variables. Write a program to read data in to these and print the same. **Refer 16.1**

**OR**

b) Write a short notes on : **(4 x 4 =16)**

( i ) Union  **Refer 16.7**

(ii) Static storage class **Refer 17.1.3**

(iii) #include statement **Refer 17.2.1**

(iv) #ifndef … #endif **Refer 17.3**