PROGRAMMING FOR PROBLEM SOLVING
(CSEN 1001)

Time Allotted : 3 hrs **Full Marks : 70**

Figures out of the right margin indicate full marks.

Candidates are required to answer Group A and any 5 (five) from Group B to E, taking at least one from each group.

Candidates are required to give answer in their own words as far as practicable.

Group – A
(Multiple Choice Type Questions)

1. Choose the correct alternative for the following: **10 × 1 = 10**

(i) The default value of a static variable is
(a) 1 (b) 0
(c) -1 (d) garbage value

(ii)
```
#include <stdio.h>
int main()
{
      int a = 3, b = 5;
    swap(a,b);
     printf("%d %d", a, b);
     return 0;
}
void swap(int a,int b)
{
  int t = a;
       a = b;
       b = t;
}
```
What will be the output?
(a) 3 5 (b) 5 3
(c) 3 3 (d) 5 5

(iii) If a two dimensional array int a[10] [20] is represented as an array of pointers, then the element a[4][5] can be denoted by:
(a) *(a + 4) + 5 (b) *a[4] + 5
(c) *(*(a + 4) + 5) (d) a[4] + 5

(iv) Which of the following is equivalent to y = y * 2? (data type of y is int).
(a) y = y << 1 (b) y = y << 2 (c) y = y >> 1 (d) y = y >> 2

(v) Find the output of the code snippet given below
```
int x;
int buf[]= {1,2,3,4,5,6,7,8,9};
x = (buf+1)[5];
printf("%d",x);
```
(a) 5 (b) 6 (c) 7 (d) 8

(vi) The expression 7 + 10 % 6 / 2 evaluates to
(a) 8 (b) 9 (c) 7 (d) 17

(vii) What would be the output of the following C code?
```
int main()
{
     char str1[] = "Hello";
     char str2[] = "Hello";
     if (str1 == str2)
          printf("\n Equal");
     else
          printf("\n Unequal");
  return 0;
}
```
(a) Equal (b) Unequal
(c) Error (d) None of these

(viii)
```
int main()
{
    int j = 10;
    for (int i=0; i<3; j -= i, i++)
    printf(" %d ",j);
    return 0;
}
```
What will be the output?

(a) 10 8 9 (b) 9 8 7

(c) 10 10 9 (d) 9 9 8

(ix) Assume a structure is declared in C, called "student". It has two fields, int rollNo and float Marks. Which of the following C statements won't generate any compilation errors?

(a) struct student sc; sc.rollNo = 123; sc.Marks = 456.0;

(b) student sc; sc.rollNo = 123; sc.Marks = 456.0;

(c) struct student *sc; sc = (student*) malloc(sizeof(student));
sc -> rollNo = 123; sc->Marks = 456.0

(d) None of the above.

(x) Which of the following functions does not use a FILE pointer as its parameter?

(a) fseek( ) (b) ftell( )

(c) rewind( ) (d) None of (a), (b) & (c).

Group- B

2. (a) The following is known for a number:
$(355)_7 = (160)_r$.
Find the value of 'r'. *[(CO1)(Understand/IOCQ)]*

(b) Describe all the steps of compilation of a C program by mentioning the intermediate files generated at each step. *[(CO2)(Remember/LOCQ)]*

(c) Compute 25 - 15 in two (binary) ways.
(i) Using the standard rule of binary subtraction.
(ii) Using 2's complement. *[(CO1)(Apply/IOCQ)]*

3 + 4 + (2.5 × 2) = 12

3. (a) Draw a Flowchart for finding out how many 1's and how many 0's are there in an integer number. *[(CO1)(Analyze/IOCQ)]*

(b) i. Convert $(1423.75)_{10}$ into its equivalent octal number.
ii. Convert $(A8F6.13B)_{16}$ into its equivalent decimal number. Show all intermediate steps. *[(CO1)(Understand/LOCQ)]*

(c) Convert the decimal number 0.125 into IEEE 754 single precision floating point representation. *[(CO1)(Apply/LOCQ)]*

4 + 4 + 4 = 12

Group - C

4. (a) What is the purpose of an external variable? What is its scope? *[(CO3)(Remember/LOCQ)]*

(b) Justify, "the range of signed char (1 byte) is -128 to +127". *[(CO3)(Understand/IOCQ)]*

(c) int a=2,b=3,c=5;
float abc = (float)((a+b+c)/3);
What is the expected output of the above C code? If the actual output is any different, please show how can it be made the same as the expected output? *[(CO3)(Analyze/IOCQ)]*

(d) Write the following using ternary operator in C:
a % b, where a is a +ve integer and b is -ve, or vice versa.
If the value is -ve, then make it positive, else double it.
What would be the output for 15 % (-4)? *[(CO3)(Analyze/IOCQ)]*

2 + 2 + 4 + 4 = 12

5. (a) Given an integer x=10. Is it possible to multiply x by 8 without using multiplication operator (*)? If yes, compare the performance of the method with that of using multiplication operator. *[(CO3, CO6)(Apply/HOCQ)]*

(b) A positive integer n > 1 is called a Mersenne prime if n is a prime number and $n = 2^k - 1$ for some positive integer k. For example, 3, 7 and 31 are all Mersenne primes. Write down a program in C to find out the largest Mersenne prime > p where p >1 is taken from keyboard. [Thus, if p = 25, your program should output 31.] *[(CO3,CO4,CO5)(Apply/HOCQ)]*

(c) Explain the output of the following programs.

```
main ( )
{
   if (5 & 2)
      printf("True");
   else
      printf("False");
   if (5 && 2)
      printf("True");
   else
      printf("False");
}
```

```
main ()
{
   int x=2, y, z ;
   x *= 3 + 2 ;
   printf ("%d\n", x);
   x *= y = z = 4;
   printf ("%d\n", x);
   x = y == z ;
   printf ("%d\n", x);
   x == ( y = z ) ;
   printf("%d\n", x);
}
```

[(CO2)(Analyze/IOCQ)]

(3 + 2) + (1 + 2) + (2 + 2) = 12

Group - D

6. (a) Define the following function with the prototype given below

int exponent(int x, int y);

This function will evaluate and return x^y if there is no overflow. Otherwise in case of overflow (i.e. value out of range for integer), the function will return 0. *[(CO3, CO5)(Apply/HOCQ)]*

(b) Write down a function in C that takes *x* and *n* as inputs, and then find the value of the following series up to *n* terms:

$$x - \frac{x^3}{2*3} + \frac{x^5}{4*5} - \frac{x^7}{6*7} + \frac{x^9}{8*9} - \cdots$$

[(CO4, CO5)(Apply/HOCQ)]

6 + 6 = 12

7. (a) Write a C function that takes a pointer to a string as parameter and replaces all spaces in the string by '-' character and returns the number of spaces replaced. *[(CO4, CO5)(Apply/HOCQ)]*

(b) Consider the following C program:

```
int main()
{
int *ptr = alloc_int();
printf("%d is the returned value \n",*ptr);
}
int *alloc_int()
{
    int ret_val = 5;
    return &ret_val;
}
```

i) What is the value printed by the "printf" statement?

ii) If this is not the desired output, how can you modify the above code suitably? *[(CO3, CO6)(Understand/IOCQ)]*

(c) Consider the following C program:

```
int main()
{
int A[5];
<unnecessary code removed>
for (int i=0; i<5;
    forFun(&i, &A[i]);
}
int forFun(int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}
```

Compare the final value in the array A[], for the following two cases when:

(i) A[ ] = {1,2,3,4,5}

(ii) A[ ] = {5,4,3,2,1}

[(CO3)(Analyze/HOCQ)]

4 + (2 + 2) + (2 + 2) = 12

Group - E

8. (a) Create a structure ***"Employee"*** having ***Name***, ***Address, Salary, and Age*** as member elements. Dynamically create an array of n (user input) 'Employee' objects to store the employee information. Then display the names of the employees having age between 40 and 50 and are living in Kolkata. *[(CO3, CO5)(Apply/HOCQ)]*

(b) Suppose a file named ***"Number.txt"*** contains a list of integers. Write a program to extract the prime numbers only from that file and write them on ***"Prime.txt"*** file. *[(CO3, CO5)(Apply/IOCQ)]*

6 + 6 = 12

9. (a) ABC company has n employees (n≥3) on its payroll. Each employee gets a secondary provident (SP) contribution from their employer besides other benefits. The % of this contribution is fixed for all employees.
   (i) Define a structure that will have Employee Name, Employee code, and Salary as the three fields. Assume any suitable datatypes for these fields.
   (ii) Create an array of employees of the above structure dynamically and populate their various fields taking input from keyboard.
   (iii) Write a program / function to print the SP contribution of all employees on a monthly basis.

[(CO3, CO5) (Understand/IOCQ)]

   (b) Write a program to delete a specified line from a text file. The filename and the line number to be deleted should be provided as command line arguments. *[(CO3, CO5)(Apply/IOCQ)]*

6 + 6 = 12

Cognition Level	*LOCQ*	*IOCQ*	*HOCQ*
Percentage distribution	*14.58*	*50*	*35.42*

Course Outcome (CO):

After the completion of the course students will be able to
CO1: Understand and remember functions of the different parts of a computer.
CO2: Understand and remember how a high-level language (C programming language, in this course) works, different stages a program goes through.
CO3: Understand and remember syntax and semantics of a high-level language (C programming language, in this course).
CO4: Understand how code can be optimized in high-level languages.
CO5: Apply high-level language to automate the solution to a problem.
CO6: Apply high-level language to implement different solutions for the same problem and analyze why one solution is better than the other.

**LOCQ: Lower Order Cognitive Question; IOCQ: Intermediate Order Cognitive Question; HOCQ: Higher Order Cognitive Question*