INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad - 500 043

COMPUTER SCIENCE AND ENGINEERING

TUTORIAL QUESTION BANK

Course Name	**COMPUTER PROGRAMMING**
Course Code	**ACS001**
Class	I B. Tech
Branch	Common for CSE / ECE / EEE / IT
Year	2017-2018
Course Coordinator	Dr. K Srinivasa Reddy, Professor and HOD, Department of IT
Team of Instructors	Dr. K Srinivasa Reddy, Dr. G Ramu, Dr. J Sirisha Devi, Dr. K Suvarchala, Ms. B Rekha, Ms. B Padmaja, Ms. G Geetha, Ms. K Laxmi Narayanamma, Mr. Y Subba Rayudu

COURSE OBJECTIVES (COs):
The course should enable the students to:

I	Learn adequate knowledge by problem solving techniques.
II	Understand programming skills using the fundamentals and basics of C Language.
III	Improve problem solving skills using arrays, strings, and functions.
IV	Understand the dynamics of memory by pointers.
V	Study files creation process with access permissions.

COURSE LEARNING OUTCOMES (CLOs):
Students, who complete the course, will have demonstrated the ability to do the following:

CACS001.01	Identify and understand the working of key components of a computer system.
CACS001.02	Analyze a given problem and develop an algorithm to solve the problem.
CACS001.03	Describe the fundamental programming constructs and articulate how they are used to develop a program with a desired runtime execution flow.
CACS001.04	Gain knowledge to identify appropriate C language constructs to write basic programs.
CACS001.05	Identify the right data representation formats based on the requirements of the problem.
CACS001.06	Describe the operators, their precedence and associativity while evaluating expressions in program statements..
CACS001.07	Understand branching statements, loop statements and use them in problem solving.
CACS001.08	Learn homogenous derived data types and use them to solve statistical problems.
CACS001.09	Understand procedural oriented programming using functions.
CACS001.10	Understand how recursion works and write programs using recursion to solve problems.

CACS001.11	Differentiate call by value and call by reference parameter passing mechanisms.
CACS001.12	Understand pointers conceptually and apply them in C programs.
CACS001.13	Distinguish homogenous and heterogeneous data types and apply them in solving data processing applications.
CACS001.14	Explain the concept of file system for handling data storage and apply it for solving problems.
CACS001.15	Differentiate text files and binary files and write the simple C programs using file handling functions.
CACS001.16	Apply the concepts to solve real-time applications using the features of C language.
CACS001.17	Possess the knowledge and skills for employability and to succeed in national and international level competitive examinations.

TUTORIAL QUESTION BANK

	UNIT – I		
	INTRODUCTION		
	PART – A (SHORT ANSWER QUESTIONS)		
S. No	**Question**	**Blooms Taxonomy Level**	**Course Learning Outcome (CLOs)**
	UNIT – I **INTRODUCTION**		
1.	**List** the two major components of a computer system?	Remember	CACS001.01
2.	**Identify** the steps in creating and running a C program?	Remember	CACS001.03
3.	**Write** the steps used in problem solving?	Remember	CACS001.02
4.	**Write** the basic set of procedures that are followed by various organizations as program development life cycle methods?	Understand	CACS001.03
5.	**State** the properties of an algorithm?	Remember	CACS001.02
6.	**Write** the parameters which effects the run time of an algorithm?	Understand	CACS001.02
7.	**State** the need for measuring the complexity of an algorithm with an example?	Understand	CACS001.02
8.	**Write** the various classes of data types ANSI C supports?	Remember	CACS001.05
9.	**State** which of the following are valid identifiers. If invalid, state the reason. a. sample1 b. data_7 c. return d. #fine e. 91-080-100 f. name & age g. _val	Understand	CACS001.05
10.	**Find** the value of *x* in the following expression? x = 3 / 2 % 6 – 3 / 9;	Understand	CACS001.06
11.	**Find** the output of following statement? printf("%s","IARE-2015"+5);	Understand	CACS001.05
12.	**Write** the size and range of the basic data types?	Remember	CACS001.05

13.	**Solve** the expression and find output of the following code?
	 void main()
	 {
	     int i = -3 , j = 2, k = 0, m;
	     m = ++i && ++j && ++k;
	     printf("%%3d%3d%3d%3d", i, j, k, m);
	 }
		Understand	CACS001.06

14.	**Find** the output of the following code?
	#include<stdio.h>
	int main()
	{
	    int a=5, b=4;
	    return (a>b)?a:b;
	}
		Remember	CACS001.06

15.	**Solve** the expression and find output of the following code?
	void main()
	{
	  int x = !5 – 4 + 2 * 5;
	  printf("%d", x);
	}
		Understand	CACS001.06

16.	**Write** the basic escape sequence characters and its meaning with example?	Remember	CACS001.06

17.	**Find** the output of c, d, e and f in the below code?
	  float c = 15/10.0;
	  int d = 15/10;
	  float e = 15/10;
	  float f = 15.0/10.0;
		Understand	CACS001.05

18.	**Find** the output of the following code?
	  int main()
	  {
	    printf("%d"+1, 123);
	    return 0;
	  }
		Understand	CACS001.05

19.	**Find** the output of the following code?
	  int main()
	  {
	    printf("%d", printf("Hi!") + printf("Bye"));
	    return 0;
	  }
		Understand	CACS001.06

20.	**Find** the output of the following code?
	  int main()
	  {
	    printf("Work" "Hard");
	    return 0;
	  }
		Understand	CACS001.06

21.	**Find** the output of the following code?
	  int main()
	  {
	    int v = 10;
	    printf("%d", v++, "%d", v- -);
	    return 0;
	  }
		Understand	CACS001.06

22.	**Find** the output of the following code?
	Note: Assume two values are entered by the user are stored in the variables v and n respectively.
	  int main()
	  {
	    int v = 5, n;
	    printf("%d",scanf("%d%d", &v, &n));
	    return 0;
	  }
		Understand	CACS001.04

23.	**Find** the output of the following code?
	  int main()
	  {
	     int a = 500, b = 100, c = 30, d = 40, e = 19;
	     a += b -= c *= d /= e %= 5;
	     printf("%2d%2d%2d%2d%2d", a, b, c, d, e);
	     return 0;
	  }
		Understand	CACS001.06

24.	**Find** the value of x, y, z for a = 9, b = 12, c = 3 (assume all are declared as float data type)
	  a. x = a – b / 3 + c * 2 – 1;
	  b. y = a – b / (3 + c) * (2 – 1);
	  c. z = a – (b / (3 + c) * 2) – 1;
		Understand	CACS001.06

25.	**Find** the output of the following code?
	  int main()
	  {
	     int a;
	     a = 015 + 0x15 + 5;
	     printf("%d", a);
	     return 0;
	  }
		Understand	CACS001.04

26.	**Find** the output of the following code?
	  int main()
	  {
	     printf("%2d%2d%2d", sizeof(3.14), sizeof(3.14f), sizeof(3.14L));
	     return 0;
	  }
		Understand	CACS001.06

27.	**Find** the output of the following code?
	  int main()
	  {
	     int a = 5;
	     a = ++i + ++i + ++i;
	     printf("%d", a);
	     return 0;
	  }
		Understand	CACS001.06

28.	**Find** the output of the following code?
	  int main()
	  {
	     int x = 025;
	     printf("Decimal = %d\n", x);
	     printf("Octal = %o\n", x);
	     printf("Hexadecimal = %x\n", x);
	  }
		Remember	CACS001.04

29.	**Find** the output of the following code? Assume y = 6 and z = 7.
	int main()
	{
	     int x = 5, y, z, p;
	     p = printf("%d\n", scanf("%d%d", &y, &z));
	     printf("x=%d \t y=%d \t z = %d \t p= %d\n", x, y, z, p);
	}
		Understand	CACS001.04

30.	**Find** the value of x and y in the following code?
	int main()
	{
	     int x, y;
	     x = sizeof("hello") - sizeof(int);
		Understand	CACS001.06

| | ```
printf("x = %d\n", x);
y = sizeof(int) - sizeof(int);
printf("y = %d", y);
}
``` | | |
| 31. | **Find** the output of the following code?<br>```
void main()
{
        int scanf = 10, getch = 20, putch;
        putch = scanf + getch;
        printf("%d", putch);
}
``` | Understand | CACS001.06 |
| 32. | **Find** the output of the following code?<br>```
void main()
{
        int i =1, j = 2;
        {
                int i =5;
                printf("%d\n", i+j);
        }
        printf("%d", i - j);
}
``` | Understand | CACS001.04 |
| 33. | **Find** the output of the following code?<br>```
void main()
{
        int x, y,z;
        x = printf("one");
        y=sizeof(printf("two"));
        z=sizeof(x += y);
        printf("%5d%5d%5d", x, y, z);
}
``` | Understand | CACS001.04 |
| 34. | **Find** the output of the following code?<br>```
void main()
{
        int x = 3, y = 4,z;
        x++;
        y-1;
        z = x + y;
        printf("%5d%5d%5d", x, y, z);
}
``` | Understand | CACS001.06 |
| 35. | **Find** the output of the following code?<br>```
void main()
{
        int x=5, y=7,z;
        z=(x==6)||(y=6);
        printf("%5d%5d%5d", x, y, z);
}
``` | Understand | CACS001.06 |
| **PART – B (LONG ANSWER QUESTIONS)** | | | |
| 1. | **Write** a program that counts from 1 to 12 and prints the count and its inversion to 5 decimal places for each count. This will require a floating point number.
1 1.00000
2 .50000
3 .33333 | Understand | CACS001.04 |

	4 .25000		
2.	**Find** out what the decimal values of the following operations are: 1. 7 & 2 2. 1 & (~1) 3. 0 & 9 4. 7 & 9 5. 1 & 7 & 9 Try to explain the results (hint: draw out the numbers as binary patterns, using the program listed)	Remember	CACS001.06
3.	The total distance travelled by a vehicle in t seconds is given by distance = ut + (at^2)/2 Where u is the initial velocity (meters per second), a is the acceleration (meters per second) . **Write** a C program to evaluate the distance travelled at regular intervals of time, given the values of u and a. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of u and a.	Understand	CACS001.04
4.	Distance between two points (x_1, y_1) and (x_2, y_2) is governed by the formula $$D^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$ **Write** a C program to compute D given the coordinates of the points.	Understand	CACS001.04
5.	Area of a triangle is given by the formula $$A = \sqrt{S(S - a)(S - b)(S - c)}$$ Where a, b and c are sides of the triangle and 2S = a + b + c. **Write** a C program to compute the area of the triangle given the values of a, b, c.	Understand	CACS001.04
6.	The price of one kg of rice is Rs. 40.75 and one kg of sugar is Rs. 30. **Write** a C program to get these values from the user and display the prices as follows. **** LIST OF ITEMS *** Item Price Rice Rs 40.75 Sugar Rs 30.00	Understand	CACS001.04
7.	**Write** a C program to read two floating point numbers using a scanf statement, assign their sum to an integer variable and then output the values of all three variables.	Understand	CACS001.04
8.	**Write** a C program to print the value 345.6789 in fixed-point format with the following specifications: a. Correct to two decimal places b. Correct to five decimal places and c. Correct to zero decimal places	Understand	CACS001.05
9.	The ABC electric company manufactures four consumer products. Their inventory position on a particular day is given below. **Code** **Quantity** **Rate(Rs.)** F105 275 575.00 H220 107 993.95 I019 321 215.50 M315 89 725.00 Write a C program to prepare the inventory report table in the following format:	Understand	CACS001.04

	INVENTORY REPORT			

	INVENTORY REPORT _____ Code Quantity Rate Value _____ --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- **Total Value:** ----------		

INVENTORY REPORT

_____

Code	Quantity	Rate	Value
---	---	---	---
---	---	---	---
---	---	---	---
---	---	---	---
		Total Value:	-----------

No.	Question	Level	Code
10.	**Write** a C program to read a four digit integer and print the sum of its digits. [Hint: use / and % operators]	Understand	CACS001.06

<p align="center"><strong style="color:red">PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)</p>

No.	Question	Level	Code
1.	Find the output of the following code? ```int main() { printf("Work" "Hard"); return 0; }```	Understand	CACS001.05
2.	```main() { float f =5.2; double d=5.2; int r= f==d; printf("result r=%d", r); }``` Analyze the above code and predict the output from printf() statement	Understand	CACS001.05
3.	```main() { printf("\nab"); printf("\bsi"); printf("\rha"); }``` Analyze the above code and predict the output from printf() statement	Understand	CACS001.05
4.	```main() { extern int i; i=4; printf("%d",i); }``` Analyze the above code and predict the output from printf() statement	Understand	CACS001.04
5.	Predict the output or error(s) for the following: ```main() { int i=-3, j=0, k=1, l=-1,p; p=++i||j++&&--k||l--; printf("result= %d",p); }```	Understand	CACS001.06
6.	Find the output of the following piece of code. ```char c[]="123sai"; printf("%d %f %s ", c, c, c);```	Understand	CACS001.05

| 7. | ```c
main()
{
    int m=-1<<4;
    printf("%d", m);
}
```
Analyze the above code and predict the output from printf() statement | Understand | CACS001.06 |
|---|---|---|---|
| 8. | ```c
#define int char
main()
{
    int p=65;
    printf("size of the variable p=%d", sizeof(p));
}
```
Analyze the above code and predict the output from printf() statement | Understand | CACS001.06 |
| 9. | Find the value of "count" at the end of the execution of the following C program.
```c
main incr (int i)
{
    static int count = 0;
    count = count + i;
    printf("%d",count);
}
``` | Understand | CACS001.04 |
| 10. | ```c
main()
{
    int p=3;
    p=!p>4;
    printf("i=%d", i);
}
```
Analyze the above code and predict the output from printf() statement | Understand | CACS001.06 |
| 11. | ```c
main()
{
    register int r;
    printf("%p\n", &r);

}
```
Analyze the above code and predict the output from printf() statement | Understand | CACS001.04 |
UNIT – II			
CONTROL STRUCTURES, ARRAYS AND STRINGS			
PART – A (SHORT ANSWER QUESTIONS)			
1.	**Find** the output of the following code?
```c
void main()
{
    int x=5;
    if(x = 6)
        printf("hello");
    else
        printf("Bye");
}
``` | Understand | CACS001.07 |
| 2. | **Find** the output of the following code?
```c
void main()
``` | Understand | CACS001.07 |

	``
	{
	    int i=5,j=6,k=7;
	    if(i<j, j>k, i==k)
	            printf("Correct");
	    else
	            printf("Wrong");
	}
	``		

3.	**Find** the output of the following code?
	``
	void main()
	{
	    int x =10, y=8, z=1;
	    if(++x || ++y)
	    {
	            printf("%5d%5d%5d", x=y, y=z, z=5);
	    }
	}
	``
		Understand	CACS001.07

4.	Take x = 0, y = 0 and z = 1. **Find** the value of x, y, and z after executing the following code?
	``
	if(x)
	    if(y)
	        z = 3;
	    else
	        z = 2;
	``
		Understand	CACS001.07

5.	**Find** the output of the following code?
	``
	int main()
	{
	    int i = 1;
	    for(; i < 4; i++);
	            printf("%d", i);
	    return 0;
	}
	``
		Understand	CACS001.07

6.	**Find** the output of the following code?
	``
	int main()
	{
	    int a, b;
	    for(a = 0; a < 10; a++);
	        for(b = 25; b > 9; b -= 3);
	            printf("%d%d", a, b);
	    return 0;
	}
	``
		Understand	CACS001.07

7.	**Find** the output of the following code?
	``
	int main()
	{
	    int a;
	    for(a = 5; --a;)
	        printf("%d", a);
	    return 0;
	}
	``
		Understand	CACS001.07

8.	**State** the difference between entry controlled and exit controlled loop with example?	Remember	CACS001.07
9.	**Write** the usage of break and continue statement with example?	Remember	CACS001.07

10.	**Find** the output of the following code?
	``
	int main()
	{
	    int a = 1, b = 2, c = 3, d = 4, e;
	``
		Understand	CACS001.07

| | ```
if(e = (a & b | c ^ d))
    printf("%d", e);
return 0;
}
``` | | |
| 11. | **Find** the output of the following code?<br>```
void main()
{
    int a=1,b=2,c=3,d=4;
    if (d > c)
        if (c > b)
            printf("%d %d", d, c);
        else if (c > a)
            printf("%d %d", c, d);
        if (c > a)
            if (b < a)
                printf("%d %d", c, a);
            else if (b < c)
                printf("%d %d", b, c);
}
``` | Understand | CACS001.07 |
| 12. | **Find** the output of the following code?<br>```
void main()
{
    int choice = 3;
    switch(choice)
    {
        default: printf("default");
        case 1: printf("choice 1"); break;
        case 2: printf("choice 2"); break;
    }
}
``` | Understand | CACS001.07 |
| 13. | **Find** the output of the following code?<br>```
void main()
{
    char c = 125;
    do
        printf("%d", c);
    while(c++);
}
``` | Understand | CACS001.07 |
| 14. | **Find** the output of the following code?<br>```
void main()
{
    for(;;)
    {
        printf("%d", 10);
    }
}
``` | Understand | CACS001.07 |
| 15. | **Find** the output of the following code?<br>```
void main()
{
    printf("hi!");
    if !(0)
        printf("bye");
}
``` | Understand | CACS001.07 |

16.	**Find** the output of the following code?
	``
	void main()
	{
	    int a =1;
	    if(a)
	            printf("test");
	    else ;
	            printf("again");
	}
	``
		Understand	CACS001.07

17.	**Find** the output of the following code?
	``
	void main()
	{
	    int i =1;
	    if(i++, ++i, i--, --i)
	            printf("%d\n", i);
	}
	``
		Understand	CACS001.07

18.	**Find** the output of the following code?
	``
	void main()
	{
	    float i;
	    for(i = 0.1;i < 0.4; i += 0.1)
	            printf("%.1f\n", i);
	}
	``
		Understand	CACS001.07

19.	**Find** the output of the following code?
	``
	void main()
	{
	    int i;
	    for(i = 2;i += 2; i <= 9; i +=2)
	            printf("%d\n", i);
	}
	``
		Understand	CACS001.07

20.	**Find** the output of the following code?
	``
	void main()
	{
	    int i = 3;
	    for(i--; i < 7; i = 7)
	            printf("%d", i++);
	}
	``
		Understand	CACS001.07

21.	**Find** errors if any from the following code?
	``
	int main()
	{
	    float x=3.5;
	    switch(x)
	    {
	            case 3.1: printf("A");
	            case 3.2: printf("B");
	            case 3.3: printf("C");
	    }
	    return 0;
	}
	``
		Understand	CACS001.07

22.	**Find** the output of the following code?
	``
	int main()
	{
	    int i=3,j=4,k=5;
	    for(++i; i==j; k++)
	    {
	            printf("hello %d", k);
	``
		Understand	CACS001.07

| | ```
        }
        return 0;
}
``` | | |
| 23. | **Find** the output of the following code?<br>```
int main()
{
        int i,j;
        for(i=1;i<3;i++)
        {
                for(j=1;j<3;j++)
                {
                        if(i==j)
                                break;
                }
        }
        printf("%5d%5d",i,j);
        return 0;
}
``` | Understand | CACS001.07 |
24.	**State** the rule that determines the order in which initial values are assigned to multi dimensional array elements?	Remember	CACS001.08

25.	**State** which of the following is the correct syntax for the initialization of one-dimensional array?
	   a. num[3]={0 0 0};
	   b. num[3]={0,0,0};
	   c. num[3]={0;0;0};
	   d. num[3]=0
		Remember	CACS001.08

26.	**State** which of the following is the correct syntax for initialization of two-dimensional array?
	   a. table[2][3]={0,0,0,1,1,1};
	   b. table[2][3]={
	              {0,0,0}
	              {1,1,1}
	          };
	   c. table[2][3]={0,1},{0,1},{0,1};
		Remember	CACS001.08

27.	**State** which of the following multi-dimensional array declaration is correct for realizing a 2x3 matrix?
	      a. int m[2][3]
	      b. int m[3][2]
	      c. int m[3],m[2]
		Remember	CACS001.08
| 28. | **Find** the output of the following code?<br>```
void main()
{
    int a[][3] = {{1, 2}, {3, 4, 5}, {5}};
    printf("%3d%3d%3d", sizeof(a), a[0][2], a[1][2]);
}
``` | Understand | CACS001.08 |
| 29. | **Write** the output of the following code?<br>```
void main()
{
    int xxx[10] = {5};
    printf("%3d%3d", xxx[1], xxx[9]);
}
``` | Understand | CACS001.08 |
| 30. | **Write** the output of the following code?<br>```
void main()
{
    int a[3][2] = {10. 20, 30, 40, 50, 60};
``` | Remember | CACS001.08 |

| | ```
    printf(“%d”, a[0][4]);
}
``` | | |
| 31. | **Distinguish** Lvalue and Rvalue of an array element?
Explain the differences with example. | Remember | CACS001.08 |
| 32. | Is it possible to pass an entire array to a function as an argument? Justify your answer with a Suitable example? | Remember | CACS001.08 |
| 33. | **Write** the output of the following code?<br>```
#include<string.h>
void main()
{
    char s1[] = "Anil kumar gupta";
    char s2[] ="kumar";
    printf(strstr(s1,s2));
}
``` | Understand | CACS001.08 |
| 34. | **Write** the output of the following code?<br>```
#include<string.h>
void main()
{
    char s1[] = "jaihind";
    char s2[] ="jaipur";
    int x;
    x =strncmp(s1,s2,3);
    printf("x = %d", x);
}
``` | Understand | CACS001.08 |
| 35. | **Write** the output of the following code?<br>```
#include<string.h>
void main()
{
  char s1[] = "NEW  DELHI";
  char s2[] ="BANGALORE";
  strncpy(s1,s2,4);
  printf("%s", s1);
}
``` | Understand | CACS001.08 |
| 36. | **State** the correct syntax for copying a string S1 into S2? | Remember | CACS001.08 |
| 37. | **Identify** which of the following is used to represent the end of a string?
    a.   Blank space
    b.   Null character
    c.   Newline character
    d.   Last element of the string | Remember | CACS001.08 |
| 38. | **Examine** the code and identify the line no containing error?<br>```
int a[10]; //line 1
int *p;    //line 2
p=a;       //line 3
a=p;       //line 4
``` | Remember | CACS001.08 |
| 39. | **Compare** the following two strings using strcmp() function and display its<br>return value?<br>```
   char x[5] = ”ABCD”;
   char y[5] = “abcd”;
``` | Remember | CACS001.08 |
| 40. | **Identify** the string function which is available in <string.h> to find the sub-string in the main string? | Understand | CACS001.08 |
| 41. | **State** various string manipulation functions in C? Write syntax and give example to each of them. | Understand | CACS001.08 |

	PART – B (LONG ANSWER QUESTIONS)		
1.	**Compare** and **Contrast** while and do while loop? Write a C program to print the odd numbers from X to Y using do while loop?	Remember	CACS001.07
2.	An electric power distribution company charges domestic consumers as follows: Consumption Units Rate of charge 0-20 Rs 0.50 per unit 201-400 Rs 100 + Rs0.65 per unit excess of 200 401-600 Rs 230 plus 0.80 per unit excess of 400 601 and above Rs 390 plus Rs 1.00 per unit excess of 600 **Write** a C program that reads the customer number and power consumed and print amount to be paid by the customer (Use else-if ladder)	Understand	CACS001.07
3.	**Write** a C program to display the traffic control signal lights based on the following. • If user entered character is R or r then print RED Light Please STOP. • If user entered character is Y or y then print YELLOW Light Please Check and Go. • If user entered character is G or g then print GREEN Light Please GO. • If user entered some other character then print THERE IS NO SIGNAL POINT.	Understand	CACS001.07
4.	Admission to a professional course is subject to the following conditions: a. Marks in Mathematics >= 60 b. Marks in Physics >= 50 c. Marks in Chemistry >= 40 d. Total in all three subjects >= 200 e. Total in Mathematics and Physics >= 150 Given the marks in the three subjects, **Write** a C program to process the application to list the eligible candidates.	Understand	CACS001.07
5.	**Write** a C program to compute the real roots of a quadratic equation $ax^2 + bx + c = 0$ The roots are given by the equations $$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$ The program should request for the values of the constants a, b and c and print the values of x1 and x2. Use the following rules: a. No solution, if both a and b are zero b. There is only one root, if a=0 c. There are no real roots, if b^2 - 4ac is negative d. Otherwise, there are two real roots **Write** a C program to test all the above conditions.	Understand	CACS001.07
6.	**Write** a program that counts from one to ten, prints the values on a separate line for each, and includes a message of your choice when the count is 3 and a different message when the count is 7.	Understand	CACS001.07
7.	**Write** a C program to calculate commission for the input value of sales amount. Commission is calculated as per the following rules: a. Commission is nil for sales amount Rs 5000/.	Understand	CACS001.07

	b. Commission is 2% for sales when sales amount is greater than 5000 and less than equal to 10000. c. Commission is 5% for sales amount greater than 10000.		
8.	A character is entered through keyboard. **Write** a C program to determine whether the character entered is a capital letter, a small case letter, a digit or a special symbol using if-else and switch case. The following table shows the range of ASCII values for various characters. Characters ASCII values A – Z 65 – 90 a – z 97 – 122 0 – 9 48 – 57 Special symbols 0 – 47, 58 – 64, 91 – 96, 123 - 127	Understand	CACS001.07
9.	If cost price and selling price of an item S input through the keyboard, write a program to determine whether the seller has made profit or incurred loss. **Write** a C program to determine how much profit or loss incurred in percentage.	Understand	CACS001.07
10.	**Write** a C program to produce the following output? 1 3 5 7 9 11 13 15 17 19	Understand	CACS001.07
11.	**Write** a C program for the following: **1.** To print the reverse of an integer number **2.** To check whether the given integer is palindrome or not.	Understand	CACS001.07
12.	**Write** a C program to print the numbers in triangular form. 1 1 2 1 2 3 1 2 3 4 1 2 3 4 5	Understand	CACS001.07
13.	**Write** a C program to read in two numbers, x and n, and then compute the sum of this geometric progression $1+x+x^2+x^3+...x^n$. For example: if n is 3 and x is 5, then the program computes 1+5+25+125. Print x, n, the sum. Perform error checking. For example the formula does not make sense for negative Exponents – if n is less than 0. Have your program print an error message if n<0, then go back and read in the nest pair of numbers of without computing the sum. Are any values of x also illegal? If so, test for them too.	Understand	CACS001.08
14.	**Write** a C program to print Armstrong numbers between 1 to n where n value is entered by the user. [Hint: Armstrong number is defined as the sum of cubes of individual digits of a number. e.g. $371 = 3^3 + 7^3 + 1^3$]	Understand	CACS001.07
15.	**Write** a C program to generate all prime numbers between 1 and n, where n value is supplied by the user.	Understand	CACS001.07

16.	**Write** a C program to print first n lines of the Pascal's Triangle. Pascal's triangle is a triangular array of the binomial coefficients.
	                  1
	             1 1
	         1 2 1
	     1 3 3 1
		Understand	CACS001.07

17.	**Write** a C program to print first n lines of Floyd's Triangle. 1 2 3 4 5 6 7 8 9 10	Understand	CACS001.07

	11 12 13 14 15		
18.	**Write** a C program to print the following series 1/1! + 2/2! + 3/3! + ……………	Understand	CACS001.07
19.	**Write** a C program to compute and display the sum of all integers that are divisible by 6 but not divisible by 4 and lie between 0 and 100. The program should also count and display the number of such values.	Understand	CACS001.07
20.	**Write** a C program to produce the following form of Floyd's triangle 1 2 3 4 5 6 7 8 9 10	Understand	CACS001.07
21.	**Write** C programs for the following: a. Find the largest and smallest number among a list of integers. b. Read a list of elements into an array and print the reverse of the list.	Understand	CACS001.08

22.	**Write** C programs for the following:
	a. Read two matrices and find the addition and multiplication of two matrices.
	b. Find the transpose of a matrix.
	          e.g. Given matrix
	                   1 2 3
	                   4 5 6
	             Transpose of the matrix:
	                   1 4
	                   2 5
	                   3 6
		Understand	CACS001.08

23.	**Write** a C program to store numbers into an array and find the frequency of a particular number in array and print it.	Understand	CACS001.08
24.	**Write** a C program to read n unsorted numbers to an array of size n and pass the address of this array to a function to sort the numbers in ascending order using bubble sort technique.	Understand	CACS001.08
25.	**Write** a C program that: **1.** Implements string copy operation STRCOPY(str1,str2) that copies a string str1 to another string str2 without using library function. **2.** Reads a sentence and prints frequency of each of the vowels and total count of consonants.	Understand	CACS001.08
26.	**Write** a C program to check whether a given matrix is sparse matrix or not. The size of the matrix must be minimum 2x2.	Understand	CACS001.08
27.	**Write** a C program to read marks obtained by a class of 50 students in subject and count the number of students belonging to each of the following group of marks: 0-9, 10-19, 20-29, 30-39, 40-49,….,100.	Understand	CACS001.08
28.	**Write** a C program accepts a string and returns true if the string is a palindrome and false if it is not, without using string built-in functions?	Understand	CACS001.08
29.	**Write** a C program to a. Check whether the given string is palindrome or not with and without using string functions. b. Insert a sub-string in to given main string from a given position.	Understand	CACS001.08
30.	**Write** a C program to a. Remove blank spaces from a string. b. Capitalize all the letters of a string.	Understand	CACS001.08
31.	**Write** a C program to accept two strings and compare them. Finally it prints whether both are equal, or first string is greater than the second or the first string is less than the second string	Understand	CACS001.08

	PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)		
1.	```
void main()
{
    int i = 5, sum = 0;
    for(i; i; i+5)
        sum = sum + i;
    printf("Sum = %d", sum);
}
```
**Analyze** the above code and predict the output from printf() statement. | Understand | CACS001.07 |
| 2. | ```
void main()
{
    int i = 5, j = 10, k = 1;
    if(++i || ++j )
        k = i + j;
    else
        k = i – j;
    printf("%3d%3d%3d", i, j, k);
}
```
Evaluate the final value of i, j, k from the above code? | Understand | CACS001.07 |
| 3. | ```
for(i = 1;i < 3; i++)
{
    for(j = 1; j < 3; j++
    {
    for(k = 1; k < 3; k++)
      {
        if(j == k)
            break;
        else
          {
            printf("%d%d%d", i,j, k);
            continue;
          }
      }
    }
}
```
**Predict** the output of the above code. | Understand | CACS001.07 |
| 4. | ```
switch (N % 6)
{
    case 3: printf("Wednesday");
    default: printf("Sunday");
    case 5:printf("Friday");
}
```
In the above code if N = 27, then **predict** the output of the code? | Understand | CACS001.07 |
| 5. | Consider the C function given below. Assume that the array listA contains n > 0 elements, sorted in ascending order.
```
int ProcessArray(int *listA, int x, int n)
{
    int i, j, k;
    i = 0;
    j = n-1;
    do
    {
        k = (i+j)/2;
        if (x <= listA[k])
            j = k-1;
``` | Understand | CACS001.08 |

	if (listA[k] <= x)
	    i = k+1;
	  }while (i <= j);
	  if (listA[k] == x)
	    return(k);
	  else
	    return -1;
	 }
	Explain the purpose of function ProcessArray?		

6.	void g(int x[10], int p)
	{
	    x[p] = p;
	    x[p – p] = p;
	}
	void main()
	{
	    int arr[3] = {10, 20, 30};
	    g(arr, 2);
	    printf("%d%d%d", arr[0], arr[1], arr[2]);
	}
	Predict the output of the above code.
		Understand	CACS001.08

7.	char a[5] = "IARE";
	int i =0;
	while(a[i])
	    printf("%s\n", (a + i++));
	Find the output of the above code.
		Understand	CACS001.08

8.	for(putchar('C');putchar('A');putchar('R'))
	putchar('T');
	Predict the output of the above code.
		Understand	CACS001.08

9.	main()
	{
	    static int i=3;
	    printf("%d",i--);
	    if(i)
	        main();
	}
		Understand	CACS001.08

<table>
<tr><td colspan="4" align="center">UNIT – III</td></tr>
<tr><td colspan="4" align="center">FUNCTIONS AND POINTERS</td></tr>
<tr><td colspan="4" align="center">PART – A (SHORT ANSWER QUESTIONS)</td></tr>
<tr><td>1.</td><td>State the advantage of user defined functions?</td><td>Remember</td><td>CACS001.09</td></tr>
<tr><td>2.</td><td>State various types of functions used in C?</td><td>Remember</td><td>CACS001.09</td></tr>
<tr><td>3.</td><td>State the difference between actual and formal parameters?</td><td>Remember</td><td>CACS001.09</td></tr>
<tr><td>4.</td><td>Write the need for a function prototype with an example?</td><td>Remember</td><td>CACS001.09</td></tr>
<tr><td>5.</td><td>State the various types of functions depending upon categories of arguments and return statements with example?</td><td>Remember</td><td>CACS001.09</td></tr>
<tr><td>6.</td><td>Define a recursive function and explain with an example?</td><td>Remember</td><td>CACS001.10</td></tr>
<tr><td>7.</td><td>Discuss the advantages and disadvantages of recursion?</td><td>Remember</td><td>CACS001.10</td></tr>
<tr><td>8.</td><td>Find the output of the following code?
void main ()
{
    static int v = 5;
    printf ("%d\t", v--);
    if(v)
        main();
}</td><td>Understand</td><td>CACS001.10</td></tr>
</table>

9.	**Write** the default return type for a function with an example?	Remember	CACS001.09

10.	**Distinguish** between the following:
	    a. Automatic and static variables
	    b. Scope and visibility of variables
		Understand	CACS001.09

11.	**Identify** the invalid prototype declarations if any with valid reasons:
	    a. int (f1) void;
	    b. void f2 (void, void);
	    c. void f3 (int a, int &b);
		Understand	CACS001.09

12.	**Find** errors if any, in the following function definitions:
	
	``c
	int abc (int a, int b)
	{
	    double c = a + b;
	    return (c);
	}
	``
		Understand	CACS001.09

13.	**Find** errors if any, in the following function calls:
	    a. xyz (int x, int y);
	    b. xyz ( ) + xyz ( );
	    c. xyz (void);
		Understand	CACS001.09

14.	**Find** the output of the following code?
	
	``c
	int prod (int m, int n);
	void main ( )
	{
	    int x = 10, y = 20, p, q;
	    p = prod (x, y);
	    q = prod(p, prod (x, y));
	    printf(“%5d%5d”, p, q);
	}
	int prod (int a, int b)
	{
	    return (a * b);
	}
	``
		Understand	CACS001.09

15.	**Find** the output of the following code?
	
	``c
	int test (int num)
	{
	        int m,n=0;
	        while(num)
	        {
	                m = num%10;
	                if(m%2)
	                        n=n+1;
	                num = num / 10;
	        }
	        return(n);
	}
	void main ( )
	{
	    int r;
	    r = test(135);
	    printf("Result = %d", r);
	}
	``
		Understand	CACS001.09

16.	**State** the reasons that is likely to happen when the following situations are encountered in a program: a. Actual parameters are less than the formal arguments in a function.	Remember	CACS001.09

	b. The order of actual parameters in the function call is different from the order of formal parameters in a function where all the parameters are of the same type.		
17.	**State** the need for dynamic memory allocation and how does it help in building complex programs?	Remember	CACS001.12
18.	**Write** the principal difference between the functions malloc() and calloc()?	Remember	CACS001.12
19.	**List out** the dynamic memory allocation functions and write its general syntax?	Remember	CACS001.12
20.	**Write** the usage of realloc () and free () function with example?	Remember	CACS001.12
21.	**Define** scope of a variable?	Remember	CACS001.12
22.	**Identify** the storage class which allows the data to be stored in CPU?	Remember	CACS001.12

23.	**Find** errors if any:
	void main ()
	{
	      extern int x = 10;
	      printf ("%d", x);
	}
		Understand	CACS001.04

24.	**Find** the output of the following code?
	extern int x;
	int x = 25;
	void main ()
	{
	      extern int x;
	      printf ("%d", x);
	}
		Understand	CACS001.04

25.	**Find** the output of the following code?
	void main()
	{
	static int i=5;
	    if(--i)
	      {
	          main();
	          printf("%d\t",i);
	      }
	}
		Understand	CACS001.04

26.	**Find** the output of the following code?
	f(int i, int j)
	{
	      i = i+j;
	       printf("%5d%5d", i, j);
	}
	void main()
	{
	      f(1,2);
	      f(2,3);
	}
		Understand	CACS001.09

27.	In C, if you pass an array as an argument to a function, **predict** what actually gets passed?	Remember	CACS001.09

28.	**Find** the output of the following code?
	  void fun()
	  {
	      static int s;
	      s = s+ 2;
	      printf("s = %d", s);
		Understand	CACS001.09

	`}`
	`void main()`
	`{`
	`    fun();`
	`    fun();`
	`}`		

29.	**Find** the output of the following code?
	`int add(int a, int b)`
	`{`
	`        int c = a+b;`
	`}`
	`void main()`
	`{`
	`        int a=10,b=20;`
	`        printf("%2d %2d %2d",a, b, add(a,b));`
	`}`
		Understand	CACS001.09

30.	**Find** the output of the following code?
	`int funct(char ch)`
	`{`
	`        ch=ch+1;`
	`        return ch;`
	`}`
	`void main()`
	`{`
	`        int a=127;`
	`        printf("%d %d", a, funct(a));`
	`}`
		Understand	CACS001.09

31.	**Write** the output of the following code?
	`int val;`
	`static int funct()`
	`{`
	`    return val*val;`
	`}`
	`void main()`
	`{`
	`    val=5;`
	`    funct();`
	`    val++;`
	`    printf("%d",funct());`
	`}`
		Understand	CACS001.09

32.	**Write** the output of the following code?
	`void main()`
	`{`
	`    void funct1(void);`
	`    void funct2(void);`
	`    clrscr();`
	`    funct1();`
	`}`
	`void funct1(void)`
	`{`
	`    printf("Ocean of ");`
	`    funct2();`
	`}`
	`void funct2(void)`
	`{`
	`    printf("Knowledge");}`
		Understand	CACS001.09

33.	**Write** the output of the following code?
	   void print(int *);
	   void print(int *);
	   void main()
	   {
	      int x=100;
	      print(&x);
	   }
	   void print(int *a)
	   {
	      printf("%d",*a);
	   }
		Understand	CACS001.12

34.	**Write** the output of the following code?
	int increment(int i)
	{
	      static int count =0;
	      count = count + 1;
	      return(count);
	}
	void main()
	{
	      int i,j;
	      for (i=0;i<=4;i++)
	             j = increment(i);
	      printf("%5d", j);
	}
		Understand	CACS001.09

35.	**Explain** the advantages of Dynamic allocation of Memory using the concept of Pointers in C.	Remember	CACS001.12
36.	**State** how a pointer variable can be declared and accessed with an example?	Remember	CACS001.12
37.	**Write** about chain of pointers and explain with example?	Remember	CACS001.12
38.	**Discuss** the disadvantages of pointers with suitable illustrations?	Remember	CACS001.12
39.	**State** the arithmetic operations which are allowed in pointers? Explain each of them with example,.	Remember	CACS001.12
40.	**What is** Dangling state? **Explain** the purpose of NULL pointer in avoiding dangling state?	Remember	CACS001.12

41.	**Find** the output of the following?
	void main()
	{
	    int n[3][2] = {3, 6, 9, 12, 15, 18};
	    printf("%2d%2d", *(n + 1)[1], **(n + 2));
	}
		Understand	CACS001.12

42.	**Find** the value of *y, *(y + 1) for the following program fragment:
	   char x [ ] = "Life is beautiful";
	   char *y = &x [3];
		Understand	CACS001.12

43.	Given int x = 10, y = 10;
	int *p1 = &x, *p2 = &y;
	Find the value of each of the following expressions:
	   a. (*p1)++
	   b. - - (*p2)
		Understand	CACS001.12

44.	**Identify** the correct expression for declaring a pointer to a function?
	   a. int (*p) (void);
	   b. int *p (void);
		Understand	CACS001.12

45.	**Find** the output of the following segment? int m[2]; *(m + 1) = 100; *m = *(m + 1); printf ("%d", m [0]);	Understand	CACS001.12

46.	**Use** void pointer to print the value of x and ch?
	    int *ip, x = 5;
	    char *cp, ch = 'a';
	    void *vp;
		Understand	CACS001.12

47.	**Write** the procedure for swapping two strings using pointers?	Remember	CACS001.12
48.	**Write** the significance of void pointer?	Remember	CACS001.12
49.	**State** the role of preprocessor?	Remember	CACS001.09
50.	**List out** the categories of preprocessor directives?	Remember	CACS001.09
51.	**Write** the different forms of macro substitution with example?	Remember	CACS001.09
52.	**State** different forms of file inclusion with example?	Remember	CACS001.09
53.	**List out** miscellaneous preprocessor directives with example?	Remember	CACS001.09
54.	**Write** the advantages of macro definitions in a program?	Remember	CACS001.09
55.	The value of a macro name cannot be changed during running of a program. **Write** your comments?	Understand	CACS001.09
56.	**Write** the need for conditional compilation and how does it help a programmer?	Remember	CACS001.09
57.	**Distinguish** between #ifdef and #if directives?	Remember	CACS001.09
58.	**Define** a macro and state how it is different from a C variable name?	Remember	CACS001.09
59.	**List out** the precautions one should take when using macros with argument?	Remember	CACS001.09
60.	**Enumerate** the differences between functions and parameterized macros?	Understand	CACS001.09
PART – B (LONG ANSWER QUESTIONS)			
1.	**Write** C programs that uses both recursive and non-recursive functions: a. Find the sum of n natural numbers b. Find the factorial of a given number	Understand	CACS001.10
2.	**Write** a C program that uses functions to do the following: a. Convert decimal number to binary number b. Convert binary number to decimal number	Understand	CACS001.09
3.	**Write** C programs that uses both recursive and non-recursive functions: a. Find the Nth Fibonacci number b. Find the reverse of a number	Understand	CACS001.10
4.	**Write** a C program that uses functions to do the following: a. Convert a Roman letter into its decimal equivalent. b. Find 2's complement of a binary number.	Understand	CACS001.09
5.	**Write** a user defined function which takes an array of sorted integers and returns the median value? [Hint: For odd set of integers there will be a single median and for even set of integers, there will be two middle values and median is the average of the two middle values]	Understand	CACS001.09

6.	**Write** the program expr, which evaluates a reverse Polish expression from the command line, where each operator or operand is a separate argument. For example, expr 2 3 4 + * Evaluates: 2 * (3+4).	Understand	CACS001.10
7.	**Define** a character array and use "strcpy" to copy a string into it. Print the string out by using a loop with a pointer to print out one character at a time. Initialize the pointer to the first element and use the double plus sign to increment the pointer. Use a separate integer variable to count the characters to print.	Understand	CACS001.08, CACS001.12
8.	Write a C function isprime(num) that accepts an integer argument and returns 1 if the argument is prime, a 0 otherwise. Write a C program that invokes this function to generate prime numbers between the given ranges.	Understand	CACS001.09
9.	**Write** a C program to find the seat position in a second class sleeper coach for the given seat number? [Hint: The sleeper coach has 72 seats and in each cabin there are 8 seats. Seat position: lower berth, upper berth, middle berth, side lower and side upper]	Understand	CACS001.09
10.	**Write** a C program to print the tomorrow's date for the given today's date. [Hint: Suppose today's date is 31st March 2016, then the next day will be 1st April 2016]	Understand	CACS001.09
11.	**Distinguish** between the following: a. Actual and formal arguments b. Scope and visibility of variables	Remember	CACS001.04
12.	**Write** a C program using function that reads an array of integers and reverses the elements of an array using pointers?	Understand	CACS001.08, CACS001.09
13.	**Write** a C program to read lines of text from the keyboard, count and display the occurrence of a particular word in that text?	Understand	CACS001.08, CACS001.09
14.	**List** out the advantages of using pointers and **explain** generic (void) pointers with a suitable example?	Remember	CACS001.12
15.	**Write** a C program that accepts a set of 5 names using array of pointers concept and displays them?	Understand	CACS001.12

16.	Given the following declarations.
	 int x=10, y=10;
	 int * P1 = &x, *P2 = &y;
	 What is the value of each of following expressions and explain why
	 (i) (*P1) ++
	 (ii) - - (*P2)
	 (iii) *P + (*P2)- -
	 (iv) ++ (*P2) - * P1
		Understand	CACS001.12

17.	**Write** a C program to pass a multi-dimensional array to a function containing marks of students and display it on the screen?	Understand	CACS001.08 CACS001.09
18.	**Write** a C program to read a list of N integers and sort it using pointers. [hint: use any sorting technique]	Understand	CACS001.12
19.	**Write** a C program to read a string and find the number of vowels, Consonants, digits and white spaces in that string?	Understand	CACS001.08 CACS001.09
20.	**Write** a C program to a. Copy the elements of one array to another array using pointers. b. Read two strings and compare these two strings character by	Understand	CACS001.08 CACS001.09

	character. Display the similar characters found in both the strings and count the number of dissimilar characters.		
21.	**Write** a C program to a. Add two numbers using pointers. b. Swap two numbers using pointers.	Understand	CACS001.09 CACS001.12
22.	**Write** a C program to a. Read the name of a person as input and prints the name in an abbreviated fashion, e.g. Ram Kumar as R K b. Read a line of text and count all occurrence of a particular word.	Understand	CACS001.08 CACS001.09
23.	**Explain** the following: a. Process of pointer initialization with an example? b. Distinguish between (*m)[5] and *m[5]?	Understand	CACS001.12
24.	**Write** a function day_name that receives a number n and returns a pointer to a character string containing the name of the corresponding day. The day names should be kept in a static table of character strings local to the function?	Understand	CACS001.08 CACS001.09

25.	Given the following declarations:
	int x = 10, y = 10;
	int *p1 = &x, *p2 = &y;
	Find the values of the following expressions:
	a. (*p1) ++
	b. - -(*p2)
	c. *p1 + (*p2) - -
	d. + +(*p2) - *p1
		Understand	CACS001.12

	PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)		

1.	**Explain** the output of the following program?
	void f(int x, int y, int z)
	{
	    printf("%d%d%d", x, y, z);
	}
	void main()
	{
	    int x = 5, y= 6, z= 7;
	    f(x = y, y = z+2, z = x+3);
	}
		Understand	CACS001.09

2.	**Analyze** the following program and identify the error in the program?
	void main()
	{
	    char ch = 'c';
	    char c = 'a';
	    char *const ptr = &ch;
	    ptr = &c;
	}
		Understand	CACS001.12

3.	**Analyze** the following program and find the output of the program?
	int fun(int a, int b)
	{
	    printf("\n a = %d", a);
	    printf("\n b = %d", b);
	}
	void main()
	{
	    int(*fptr)(int,int);
	    fptr = func;
	    func(2, 3);
	    fptr(2,3);}
		Understand	CACS001.12

4.	**Analyze** the following program and find the output of the program?
	char s[100];
	char *fun(char s[])
	{
	    static int i = 0;
	    if(*s)
	    {
	        fun(s + 1);
	        s[i] = *s;
	        i++;
	    }
	    return s;
	}
	void main()
	{
	    char s[] = "sample code";
	    printf("%s", fun(s));
	}
		Understand	CACS001.12

5.	**Analyze** the following program and find the output of the program?
	void main()
	{
	    char s1[7] = "1234", *p;
	    p = s1 + 2;
	    *p = '\0';
	    printf("%s", s1);
	}
		Understand	CACS001.12

6.	Consider the following three C functions :,
	[PI] int * g (void)
	      {
	          int x = 10;
	          return (&x);
	      }
	[P2] int * g (void)
	      {
	          int * px;
	          *px = 10;
	          return px;
	      }
	[P3] int *g (void)
	      {
	          int *px;
	          px = (int *) malloc (sizeof(int));
	          *px = 10;
	          return px;
	      }
	Identify which of the above three functions are likely to cause
	problems with pointers?
	a. Only P3
	b. Only P1 and P3
	c. Only P1 and P2
	d. P1, P2 and P3
		Understand	CACS001.12

7.	Find the output of the following C program?
	int f(int x, int *py, int **ppz)
	{
	    int y, z;
	    **ppz += 1;
		Understand	CACS001.12

| | ```
        z = **ppz;
         *py += 2;
         y = *py;
         x += 3;
         return x + y + z;
    }
    void main()
    {
         int c, *b, **a;
         c = 4;
         b = &c;
         a = &b;
         printf( "%d", f(c,b,a));
         getchar();
    }
``` | | |
| 8. | Consider the C program shown below. Find the output of this program code?<br>```
# define print(x) printf ("%d", x)
int x;
void Q(int z)
{
     z += x;
     print(z);
}
void P(int *y)
{
     int x = *y+2;
     Q(x);
     *y = x-1;
     print(x);
}
main(void)
{
     x=5;
     P(&x);
     print(x);
     getchar();
}
``` | Understand | CACS001.12 |
| 9. | Consider the following C program<br>```
main()
{
     int x, y, m, n;
     scanf ("%d %d", &x, &y);   /* x > 0 and y > 0 */
     m = x; n = y;
     while (m != n)
     {
         if(m>n)
                m = m - n;
         else
                n = n - m;
     }
     printf("%d", n);
}
```<br>The program computes<br>a.      x+y using repeated subtraction<br>b.      x mod y using repeated subtraction | Understand | CACS001.07 |

	c. the greatest common divisor of x and y d. the least common multiple of x and y		

10.	Predict the output of the following code?
	double foo (double); /* Line 1 */
	int main ()
	{
	    double da, db;
	   // input da
	   db = foo (da);
	}
	double foo (double a)
	{
	    return a;
	}
		Understand	CACS001.09

11.	char *foo()
	{
	   char *start = "hello";
	   char *end = start + 5;
	   return (start + end) / 2;
	}
	
	Analyze the piece of code and predict the return value.
		Understand	CACS001.12

12.	char foo()
	{
	   char hello[] = "hello";
	   char *foo = hello;
	   return (foo);
	}
	Analyze the piece of code and predict the return value.
		Understand	CACS001.12

UNIT – IV
STRUCTURES AND UNIONS
PART – A (SHORT ANSWER QUESTIONS)

1.	**Define** a structure and state how the members of a structure are accessed with example?	Remember	CACS001.13
2.	**Write** the major differences between arrays and structures?	Remember	CACS001.13
3.	**Write** an example of nested structure?	Remember	CACS001.13
4.	**State** the difference between a structure and union?	Remember	CACS001.13
5.	**Write** an example of array of structures?	Remember	CACS001.13
6.	**Write** the general format of sending a copy of a structure to the called function?	Remember	CACS001.13

7.	The uninitialized integer data type of a structure contains which of the following default values
	    a. Garbage
	    b. Zero
	    c. One
		Remember	CACS001.13

8.	**Identify** the following expressions which are correct for accessing the 'num' variable value of the ith element of a structure array 'student'
	    a. student[i].num
	    b. student.num[i]
	    c. student[i]->num
		Remember	CACS001.13

9.	**Find** the output of the following?
	
	``c
	struct
	{
	    int i;
	    float f;
	}var;
	void main()
	{
	    var.i=5;
	    var.f=9.76723;
	    printf("%d %.2f",var.i,var.f);
	}
	``
		Understand	CACS001.13

10.	**Write** the output of the following?
	
	``c
	struct values
	{
	    int i;
	    float f;
	};
	void main()
	{
	    struct values var={555,67.05501};
	    printf("%2d %.2f",var.i,var.f);
	}
	``
		Understand	CACS001.13

11.	**Write** the output of the following?
	
	``c
	union A
	{
	    char ch;
	    int i;
	    float f;
	}temp;
	void main()
	{
	    temp.ch='A';
	    temp.i=777;
	    temp.f=12345.12345;
	    printf("%d", temp.i);
	}
	``
		Understand	CACS001.13

12.	**Write** the output of the following?
	
	``c
	void main()
	{
	    struct employee
	    {
	        unsigned id: 8;
	        unsigned sex:1;
	        unsigned age:7;
	    };
	    struct employee emp1={203,1,23};
	    printf("%d\t%d\t%d",emp1.id,emp1.sex,emp1.age);
	}
	``
		Understand	CACS001.13

13.	**Write** an example for enumerated data type?	Remember	CACS001.13
14.	**State** the default starting value of enumerated set?	Remember	CACS001.13
15.	**Write** the usage of typedef with example?	Remember	CACS001.13

16.	**Write** the value of tulip from the following enumerated flowers? enum flowers{rose, lily = 5, lotus, tulip, sunflower);	Remember	CACS001.13
17.	**State** the operator which connects the structure name to its member name?	Remember	CACS001.13

18.	Size of a union is determined by size of the.
	        a. First member in the union
	        b. Last member in the union
	        c. Biggest member in the union
	        d. Sum of the sizes of all members
		Remember	CACS001.13

19.	**Find** the size of the following union declaration?
	union Temp
	{
	        double a;
	        int b[10];
	        char c;
	}u;
	(Assuming size of double = 8, size of int = 4, size of char = 1)
		Understand	CACS001.13

20.	Bit fields can only be declared as part of a structure
	        a. false
	        b. true
	        c. can't say
	        d. none
		Understand	CACS001.13

	PART – B (LONG ANSWER QUESTIONS)		
1.	**Write** a C program to read your full name, Date of birth and display the same using the concept of nested structure.	Understand	CACS001.13
2.	**Write** a C program to maintain a book structure containing name, author and pages as structure members. Pass the address of structure variable to a user defined function and display the contents.	Understand	CACS001.13
3.	A marketing company is having 50 employees and it maintains employee records in terms of their empid, empname, desg, salary, quantity, sales amount. The company gives 10% hike in salary to the employees if their sales amount is more than 50000/-. **Write** a C program that displays the employee records who got hike in salary.	Understand	CACS001.13
4.	IARE College is maintaining student attendance records by storing rollno, stdname, attendance percentage in 5 different subjects. **Write** a C program to find the average attendance percentage and print the following a. If attendance percentage >=75 then print student is eligible for writing final exam. b. If attendance percentage >= 65 and <75 then print student is in condonation list. c. Otherwise not eligible for writing exams.	Understand	CACS001.13

5.	Consider the declaration of the structure
	        typedef struct
	        {
	          char x;
	          char *y;
	          int z[20];
	        } status;
	Discuss whether the following are valid, if invalid, give reason.
	a. struct status s1;
	b. struct status s2[25];
	c. status s3;
		Understand	CACS001.13

	d. status s4 [20];		
6.	**Compare and Explain** the following with suitable examples: a. Nested Structures b. Array of structures	Understand	CACS001.13
7.	**Explain** the following with suitable example: a. self referential structures b. enumerated types	Remember	CACS001.13
8.	**Write** a C program to pass a copy of the entire structure named 'stores' containing members product-name, price and quantity to a function?	Understand	CACS001.13
9.	**Compare** Unions and Structures . **Explain** the differences with examples.	Remember	CACS001.13
10.	**What are** different ways of assigning values to structure members? **Explain** each method with examples.	Remember	CACS001.13
11.	**Explain** three different approaches that can be used to pass structures as function arguments. Illustrate each of them with suitable Example.?	Remember	CACS001.13
12.	**Define** a structure called complex consisting of two floating point numbers x and y and declare a variable p of type complex. Assign initial values 0.0 and 1.1 to the members.	Understand	CACS001.13
13.	**Define** a structure data type called time_struct containing 3 members integer hour, integer minute and integer second. Develop a program that would assign values to the individual members and display the time in the following format: **16 : 40 : 51**	Understand	CACS001.13
14.	**Define** a structure named census with the following 3 members: a. A character array city[ ] to store names. b. A long integer to store population of the city. c. A float member to store the literacy level. **Write** a program to do the following: a. To read details for 5 cities randomly using an array variable. b. To sort the list alphabetically. c. To sort the list based on literacy level. d. To sort the list based on population. e. To display sorted lists.	Understand	CACS001.13
15.	**Define** a structure that can describe a hotel. It should have members that include the name, address, grade, average room charge, and number of rooms. **Write** functions to perform the following operations: a. To print out hotels of a given grade in order of charges. b. To print out hotels with room charges less than a given value.	Understand	CACS001.13
16.	**Define** a structure called cricket that will describe the following information: Player name Team name Batting average Using cricket, declare an array player with 50 elements and **write** a program to read the information about all the 50 players and print a team-wise list containing names of players with their batting average.	Understand	CACS001.13
17.	**Define** a 'slack byte'? **Explain** how it affects the implementation of structures through sample code.	Remember	CACS001.13
18.	**Explain** the meaning and purpose of the following: a. struct keyword b. typedef keyword	Understand	CACS001.13

	c. sizeof operator		
19.	**Compare and contrast** structures and unions. **Write** a C program to maintain a record of 'n' student details using an array of structures with four fields (roll no, name, marks and grade). Assume appropriate data type for each field. Print the marks of the student name as input.	Understand	CACS001.13
20.	IARE maintains salary details of every employee by storing their name, department, basic pay, da, hra and cca. Store this information in a nested structure and display the salary of an employee.	Understand	CACS001.13

21.	Given the following structure and variable definitions,
	struct customer
	 {
	    char lastName[15];
	    char firstName[15];
	    int customerNumber;
	     struct
	      {
	       char phoneNumber[11];
	       char address[50];
	       char city[15];
	       char state[3];
	       char zipCode[6];
	      } personal;
	    } customerRecord, *customerPtr;
	customerPtr = &customerRecord;
	Write an expression that can be used to access the structure member in each of the following parts:
	a) Member lastName of the structure pointed to by customerPtr.
	b) Member phoneNumber of member personal of structure customerRecord.
	c) Member phoneNumber of member personal of the structure pointed to by customerPtr.
	d) Member zipCode of member personal of the structure pointed to by customerPtr.
		Understand	CACS001.13

22.	A bookshop uses a personal computer to maintain the inventory of books that are being sold at the shop. The list includes details such as author, title, isbn number, price, author, stock position. Whenever a customer wants a book, the shopkeeper inputs the title or isbn number and the system replies whether the book is available or not. If it is not, an appropriate message is displayed. If book is in the list, then the system displays the book details and asks for number of copies. If the requested copies are available, the total cost of the books is displayed, otherwise the message "Requested copies are not in stock" is displayed. Implement using structures.	Understand	CACS001.13
23.	Declare a calendar as an array of 366 elements. Each element of the array is a structure having three fields. The first field is the name of the month (a dynamically allocated string), the second field is the day of the month (an integer). The third field is the description of the activities for a particular day (a dynamically allocated string).	Understand	CACS001.13
24.	Define a structure called cricket that will describe the following information: Player name, team name, batting average. Using cricket, declare an array player with 10 elements and write a program to read the information about all the 50 players and print a team wise list containing names of players with their batting average.	Understand	CACS001.13
	PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)		

1.	**Analyze** the following program and find out the error in the program?
	`#include<stdio.h>`
	`int main()`
	`{`
	    `struct a`
	    `{`
	      `float category:5;`
	      `char scheme:4;`
	    `};`
	    `printf("size=%d", sizeof(struct a));`
	    `return 0;`
	`}`
		Understand	CACS001.13

2.	**Predict** the output of the program?
	`#include<stdio.h>`
	`int main()`
	`{`
	    `struct value`
	    `{`
	      `int bit1:1;`
	      `int bit3:4;`
	      `int bit4:4;`
	    `}bit={1, 2, 13};`
	    `printf("%d, %d, %d\n", bit.bit1, bit.bit3, bit.bit4);`
	    `return 0;`
	`}`
		Understand	CACS001.13

3.	**Verify** the following statements which correctly assigns 12 to month using
	pointer variable pdt?
	`#include<stdio.h>`
	`struct date`
	`{`
	    `int day;`
	    `int month;`
	    `int year;`
	`};`
	`int main()`
	`{`
	    `struct date d;`
	    `struct date *pdt;`
	    `pdt = &d;`
	    `return 0;`
	`}`
		Understand	CACS001.13

4.	**Predict** the output of the program?
	`#include<stdio.h>`
	`int main()`
	`{`
	    `enum days {MON=-1, TUE, WED=6, THU, FRI, SAT};`
	    `printf("%d, %d, %d, %d, %d, %d\n", MON, TUE, WED, THU, FRI,`
	    `SAT);`
	    `return 0;`
	`}`
		Understand	CACS001.13

5.	**Analyze** the program and identify the error in the program?
	`#include<stdio.h>`
	`int main()`
	`{`
	    `struct emp`
	    `{`
	        `char name[25];`
	        `int age;`
	        `float bs;`
	    `};`
	    `struct emp e;`
	    `e.name = "suresh";`
	    `e.age = 25;`
	    `printf("%s %d\n", e.name, e.age);`
	    `return 0;`
	    `}`
		Understand	CACS001.13

6.	**Analyze** the code and identify the statements which are correct in the following program?
	`#include<stdio.h>`
	`int main()`
	`{`
	    `union a`
	    `{`
	        `int i;`
	        `char ch[2];`
	    `};`
	    `union a u1 = {512};`
	    `union a u2 = {0, 2};`
	    `return 0;`
	`}`
	a. u2 CANNOT be initialized as shown.
	b. u1 can be initialized as shown.
	c. To iniatialize char ch[] of u2 '.' Operator should be used.
	d. The code causes an error 'Declaration syntax error'
		Understand	CACS001.13

7.	`struct student`
	  `{`
	    `char *name;`
	  `};`
	  `void main()`
	  `{`
	    `struct student s, m;`
	    `s.name = "st";`
	    `m = s;`
	    `printf("%s%s", s.name, m.name);`
	  `}`
	Analyze the above code and predict the output from printf() statement
		Understand	CACS001.13

8.	`Struct`
	  `{`
	    `int foo, bar;`
	`} baz;`
	`int *example()`
	`{`
	    `return &baz.foo;`
	`}`
	Analyze the above code and predict the value of return statement.
		Understand	CACS001.13

	UNIT – V		
	FILES		
	PART – A (SHORT ANSWER QUESTIONS)		
1.	**Write** the basic operations of a file?	Understand	CACS001.13
2.	**Write** the various text file opening modes?	Remember	CACS001.15
3.	**State** the various types of status enquiry library functions in C?	Remember	CACS001.15
4.	**Write** the syntax and usage of ftell()?	Remember	CACS001.15
5.	**Write** the purpose of fseek() with example?	Remember	CACS001.15
6.	**Write** the syntax and usage of rewind()?	Remember	CACS001.15

7.	**Find** the output of the following
	int main()
	{
	  FILE *fp = stdin;
	  int n;
	  fprintf(fp, "%d", 45);
	}
		Understand	CACS001.14

8.	If there is any error while opening a file, fopen() will return?
	  a. Nothing
	  b. EOF
	  c. NULL
	  d. Depends on compiler
		Understand	CACS001.15

9.	**Find** the meaning of 'a' in the following operation? fp = fopen("sample.txt", "a");	Understand	CACS001.15

10.	**Identify** which is correct about a FILE
	  a. A structure tag declared in stdio.h
	  b. One of the basic data types in c
	  c. Pointer to the structure defined in stdio.h
	  d. It is a type name defined in stdio.h
		Remember	CACS001.15

11.	**Predict** the output of this code?
	#include <stdio.h>
	int main()
	{
	  FILE *fp = stdout;
	  stderr = fp;
	  fprintf(stderr, "%s", "hello");
	}
		Understand	CACS001.15

12.	**Find** the output of this code?
	#include <stdio.h>
	#include <stdlib.h>
	int main()
	{
	  FILE *fp = stdout;
	  int n;
	  fprintf(fp, "%d", 45);
	}
		Understand	CACS001.14

13.	**Find** which is true about stdout, stdin and stderr?
	  a. File pointers
	  b. File descriptors
	  c. Streams
		Remember	CACS001.14

	d. Structure		

14.	**Predict** the output of this code?
	#include <stdio.h>
	#include <string.h>
	int main()
	{
	  char line[3];
	  fgets(line, 3, stdin);
	  printf("%d\n", strlen(line));
	  return 0;
	}
		Understand	CACS001.15

15.	**Find** the content of 'file.c' after executing the following program?
	#include<stdio.h>
	int main()
	{
	  FILE *fp1, *fp2;
	  fp1=fopen("file.c", "w");
	  fp2=fopen("file.c", "w");
	  fputc('A', fp1);
	  fputc('B', fp2);
	  fclose(fp1);
	  fclose(fp2);
	  return 0;
	}
		Understand	CACS001.15

16.	If the file 'source.txt' contains a line "Be my friend", **predict** the output of below program?
	#include<stdio.h>
	int main()
	{
	  FILE *fs, *ft;
	  char c[10];
	  fs = fopen("source.txt", "r");
	  c[0] = getc(fs);
	  fseek(fs, 0, SEEK_END);
	  fseek(fs, -3L, SEEK_CUR);
	  fgets(c, 5, fs);
	  puts(c);
	  return 0;
	}
		Understand	CACS001.15

17.	**Identify** the error in the program?
	#include<stdio.h>
	#include<stdlib.h>
	int main()
	{
	  unsigned char;
	  FILE *fp;
	  fp=fopen("trial", "r");
	  if(!fp)
	   {
	    printf("Unable to open file");
	    exit(1);
	   }
	  fclose(fp);
	  return 0;
	}
		Understand	CACS001.15

18.	**Identify** which is true about fseek() ? fseek() should be preferred	Remember	CACS001.14

	over rewind() mainly because
	   a. rewind() doesn't work for empty files
	   b. rewind() may fail for large files
	   c. In rewind, there is no way to check if the operations completed successfully
	   d. All of the above		

19.	When fopen() is not able to open a file, it returns
	   a. EOF
	   b. NULL
	   c. Runtime Error
	   d. Compiler Dependent
		Remember	CACS001.14

20.	**Identify** which of the following is true about FILE *fp
	   a. FILE is a keyword in C for representing files and fp is a variable of FILE type.
	   b. FILE is a structure and fp is a pointer to the structure of FILE type
	   c. FILE is a stream
	   d. FILE is a buffered stream
		Remember	CACS001.14

<td colspan="3" align="center">**PART – B (LONG ANSWER QUESTIONS)**</td>			
1.	**Write** a C program to read a text file containing some paragraph. Use fseek() function and read the text after skipping 'n' characters from beginning of the file?	Understand	CACS001.14

2.	**Explain** the following functions through a sample program which reads a file 'test.txt' .
	   a. ftell()
	   b. fseek()
	   c. rewind()
		Understand	CACS001.14

3.	**Write** a C program to read a text file "sample.txt" and print the following.
	   a. Substring of N characters from the position I.
	   b. Reverse order of substring of N characters produced in a.
		Understand	CACS001.14

4.	**Write** the syntax of the following file I/O functions and **Explain** Every option in each function with suitable example :
	   a. fopen()
	   b. fclose()
	   c. fread()
	   d. fwrite()
		Understand	CACS001.14

5.	**Write** a C program to open a file names INVENTORY and store in it the following data **Item** **number** **price** **quantity** Printer P100 7500 10 Scanner S200 5500 5 Hard disk H300 4500 8 Read the data from the INVENTORY file and display the inventory table with the value of each item. [Hint: value = price * quantity and use fprintf() and fscanf() functions]	Understand	CACS001.15
6.	**Write** a C program to read a given file, convert first letter of each word into uppercase and copy the contents of converted file into a new file.	Understand	CACS001.15
7.	**Write** a C program to read name and marks of 'n' number of students from user and store them in a file. If the file previously exists, then add the information of n students to the end of existing content.	Understand	CACS001.14

8.	**Write** a C program to print the following from a given file: 1. Number of characters 2. Number of spaces 3. Number of tabs 4. Number of newlines	Understand	CACS001.14
9.	Create a structure named employee containing name, age and basic pay. **Write** a C program to create 5 employee records and write to a file. Then read the records from file and display it.	Understand	CACS001.14
10.	**Write** a C program to maintain a record of "n" student details using an array of structures with four fields (Roll number, Name, Marks, and Grade). Each field is of an appropriate data type. Print the marks of the student given student name as input.	Understand	CACS001.14
	PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)		
1.	In fopen(), the open mode "wx" is sometimes preferred "w" because. 1) Use of wx is more efficient. 2) If w is used, old contents of file are erased and a new empty file is created. When wx is used, fopen() returns NULL if file already exists. a. Only 1 b. Only 2 c. Both 1 and 2 d. Neither 1 and 2	Understand	CACS001.15
2.	**Write** a C program that request for a file name and an integer known as offset value. The program then reads the file starting from the location specified by the offset value and prints the contents on the screen. If the offset value is a positive integer then printing skips that many lines. If it is negative number it prints that many lines from the end of the file. An appropriate error message should be printed if anything goes wrong.	Apply	CACS001.16
3.	**Write** a menu driven C program to add, display, search, update and delete the student record. Every student record contains name, roll no, age and marks in individual subjects.	Apply	CACS001.16
4.	**Write** a function that, given a binary file, copies the odd items (items 1,3,5, …, n) to a second binary file and the even items (items 2,4,6, … , n) to a third binary file. After all items have been copied, print the contents of both output files.	Apply	CACS001.16

Prepared by   **:** Dr. J Sirisha Devi, Dr. K Suvarchala
Date   **:** 02-09-2017

<div align="right">**HOD, CSE**</div>