



All Programs Semester I CAT II – January 2022

Answer uploading Template

Enrolment / Admission No. of Student	21SCSE1011615	Name of Course	Programming for problem solving - C
Name of Student	Abhinav kumar choudhary	Course Code	BCS01T1003
Program	BTech. CSE	Date of Examination	16/02/21
Semester	First	Time	8:00 – 9:30
Signature of Student	Abhinav kumar chy		

Student shall start writing from below:

- 1) The rules that apply to a function call are:-
- (i) Write the name of the function.
 - (ii) Add parentheses () after the function's name.
 - (iii) Inside the parenthesis, add any parameters that the function
 - (iv) End the line with a semicolon.

2) We can assign a structure (or class in C++ only) variable to another variable of same type. When we assign a structure variable to another, all members of the variable are copied to the other structure variable.

```
3) #include <stdio.h>
```

```
#include <math.h>
```

```
int main()
```

```
{
```

```
printf("Enter a number to find factorial:");
```

```
printf("\n Factorial of a given Number = %", fact());
```

```
return 0;
```

```
}
```

```
int fact()
```

```
{
```

```
int i, fact=1, n;
```

```
scanf("%d", &n);
```

```
for (i=1; i <= n; i++)
```

```
{
```

```
fact = fact * i;
```

```
}
```

```
return fact;
```

```
}
```

Output:-

Enter a number to find factorial: 5

Factorial of a given number is: 120

4)

```
#include <stdio.h>
```

```
#include <string.h>
```

```
struct student
```

```
{
```

```
    int id;
```

```
    char name[20];
```

```
    float percentage;
```

```
};
```

```
void func(struct student record);
```

```
int main()
```

```
{
```

```
    struct student record;
```

```
    record.id = 1;
```

```
    strcpy(record.name, "Raju Raj");
```

```
    record.percentage = 92.8;
```

```
    func(record);
```

```
    record = 0;
```

```
}
```

```
void func(struct student record)
```

```
    printf("Id is: %d\n", record.id);
```

```
    printf("Name is: %s\n", record.name);
```

```
    printf("Percentage is: %f\n", record.percentage);
```

```
}
```

Output :-

Id is: 1

Name is: Raju Raj

Percentage: 92.8


```

5) #include <stdio.h> // include stdio.h
#define ROW 2
#define COL 3
int main()
{
    int i, j, arr1[ROW][COL], arr2[ROW][COL];
    printf("Enter first matrix: \n");
    for(i=0; i<ROW; i++)
    {
        for(j=0; j<COL; j++)
        {
            scanf("%d", &arr1[i][j]);
        }
    }
    printf("\nEnter second matrix: \n");
    for(i=0; i<ROW; i++)
    {
        for(j=0; j<COL; j++)
        {
            scanf("%d", &arr2[i][j]);
        }
    }
    printf("\narr1 + arr2 = \n");
    // add two matrices
    for(i=0; i<ROW; i++)

```

P.T.O

5 →

```
{  
Print for (J = 0; J < COL; J++)  
{  
    printf("%5d", arr1[i][J] + arr2[i][J]);  
}  
    printf("\n")  
}  
return 0;  
}
```

Expected Output

① 1 3 6 [First matrix]
4 6 7

② 2 5 8 [Second matrix]
10 12 16

arr1 + arr2 =

3 8 14
14 18 23

- 6) There are three ways by which the values of structure can be transferred from one function to another. They are as follows-
- * Passing individual members as arguments to function.
 - * Passing entire structure as an argument to function.
 - * Passing the address of structure as an argument to function.

6) → Now let's see how to pass an individual member of structure elements as argument to function.

- Each member is passed as an argument in the function call.
- They are collected independently in ordinary variables in function header.

Example

```
#include <stdio.h>

struct date {
    int day;
    int mon;
    int yr;
};

main() {
    struct date d = {15, 09, 2021}; // struct date d;
    display(d.day, d.mon, d.yr); // passing individual mem as argument
    getch()                      to function
}

display(int a, int b, int c) {
    printf("day = %d\n", a);
    printf("month = %d\n", b);
    printf("year = %d\n", c);
}
```

Output

day - 15

month - 09

year - 2021