

## LAB ASSIGNMENT 2

1. Create a structure called "Student" with members name, age and total marks. Write a C program to input data for two students, display their information and find the average of total marks

main.c	Output
<pre>1 #include &lt;stdio.h&gt; 2 3 struct student { 4     char name[50]; 5     int age; 6     int total_marks; 7 }; 8 9 int main() { 10     struct student s1, s2; 11     float average; 12 13     printf("Enter details for student 1:\n"); 14     printf("Name: "); 15     fgets(s1.name, sizeof(s1.name), stdin); 16     printf("Age: "); 17     scanf("%d", &amp;s1.age); 18     printf("Total Marks: "); 19     scanf("%d", &amp;s1.total_marks); 20     getchar(); 21 22     printf("\nEnter details for student 2:\n"); 23     printf("Name: "); 24     fgets(s2.name, sizeof(s2.name), stdin); 25     printf("Age: "); 26     scanf("%d", &amp;s2.age); 27     printf("Total Marks: "); 28     scanf("%d", &amp;s2.total_marks); 29 30     printf("\nStudent 1 Information:\n"); 31     printf("Name: %sAge: %d\nTotal Marks: %d\n", s1.name, s1.age,         s1.total_marks);</pre>	<pre>Enter details for student 1: Name: Priyanka Age: 19 Total Marks: 87  Enter details for student 2: Name: Vijaya Age: 18 Total Marks: 89  Student 1 Information: Name: Priyanka Age: 19 Total Marks: 87  Student 2 Information: Name: Vijaya Age: 18 Total Marks: 89  Average Total Marks of both students: 88.00  === Code Execution Successful ===</pre>

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

```
struct student {
    char name[50];
    int age;
    int total_marks;
};
```

```
int main() {
    struct student s1, s2;
    float average;

    printf("Enter details for student 1:\n");
    printf("Name: ");
    fgets(s1.name, sizeof(s1.name), stdin);
    printf("Age: ");
    scanf("%d", &s1.age);
    printf("Total Marks: ");
    scanf("%d", &s1.total_marks);
    getchar();
```

```

printf("\nEnter details for student 2:\n");
printf("Name: ");
fgets(s2.name, sizeof(s2.name), stdin);
printf("Age: ");
scanf("%d", &s2.age);
printf("Total Marks: ");
scanf("%d", &s2.total_marks);

printf("\nStudent 1 Information:\n");
printf("Name: %sAge: %d\nTotal Marks: %d\n", s1.name, s1.age, s1.total_marks);

printf("\nStudent 2 Information:\n");
printf("Name: %sAge: %d\nTotal Marks: %d\n", s2.name, s2.age, s2.total_marks);

average = (s1.total_marks + s2.total_marks) / 2.0;

printf("\nAverage Total Marks of both students: %.2f\n", average);

return 0;
}

```

2.Design a structure named “Car” to store details like ID,model and rental rate per day.Write a C program to input data for three cars,calculate the total rental cost for a specified number of days,and display the results.

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

```
// Structure definition
```

```

struct Car {
    int ID;
    char model[50];
    float rental_rate_per_day;
};

```

```

int main() {
    struct Car car1, car2, car3;
    int days;
    float total_rental_cost1, total_rental_cost2, total_rental_cost3
    printf("Enter details for Car 1:\n");
    printf("Car ID: ");
    scanf("%d", &car1.ID);
    getchar();
    printf("Model: ");
    fgets(car1.model, sizeof(car1.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car1.rental_rate_per_day);

    printf("\nEnter details for Car 2:\n");

```

```
printf("Car ID: ");
scanf("%d", &car2.ID);
getchar();
printf("Model: ");
fgets(car2.model, sizeof(car2.model), stdin);
printf("Rental Rate per Day: ");
scanf("%f", &car2.rental_rate_per_day);
```

```
printf("\nEnter details for Car 3:\n");
printf("Car ID: ");
scanf("%d", &car3.ID);
getchar();
printf("Model: ");
fgets(car3.model, sizeof(car3.model), stdin);
printf("Rental Rate per Day: ");
scanf("%f", &car3.rental_rate_per_day);
printf("\nEnter the number of days for rental: ");
scanf("%d", &days);
```

```
total_rental_cost1 = car1.rental_rate_per_day * days;
total_rental_cost2 = car2.rental_rate_per_day * days;
total_rental_cost3 = car3.rental_rate_per_day * days;
```

main.c

Share

Run

Output

Clear

```

1 #include <stdio.h>
2
3 struct Car {
4     int ID;
5     char model[50];
6     float rental_rate_per_day;
7 };
8
9 int main() {
10     struct Car car1, car2, car3;
11     int days;
12     float total_rental_cost1, total_rental_cost2,
        total_rental_cost3;
13
14     printf("Enter details for Car 1:\n");
15     printf("Car ID: ");
16     scanf("%d", &car1.ID);
17     getchar();
18     printf("Model: ");
19     fgets(car1.model, sizeof(car1.model), stdin);
20     printf("Rental Rate per Day: ");
21     scanf("%f", &car1.rental_rate_per_day);
22
23     printf("\nEnter details for Car 2:\n");
24     printf("Car ID: ");
25     scanf("%d", &car2.ID);
26     getchar();
27     printf("Model: ");
28     fgets(car2.model, sizeof(car2.model), stdin);
29     printf("Rental Rate per Day: ");
30     scanf("%f", &car2.rental_rate_per_day);
31
32     printf("\nEnter details for Car 3:\n");
33     printf("Car ID: ");
34     scanf("%d", &car3.ID);
35     getchar();
36     printf("Model: ");
37     fgets(car3.model, sizeof(car3.model), stdin);
38     printf("Rental Rate per Day: ");
39     scanf("%f", &car3.rental_rate_per_day);
40
41     printf("\nEnter the number of days for rental: ");
42     scanf("%d", &days);
43
44     total_rental_cost1 = car1.rental_rate_per_day * days;
45     total_rental_cost2 = car2.rental_rate_per_day * days;
46     total_rental_cost3 = car3.rental_rate_per_day * days;
47
48     printf("\nCar Rental Details:\n");
49
50     printf("\nCar 1 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car1.ID, car1.model, car1.rental_rate_per_day, days, total_rental_cost1);
51
52     printf("\nCar 2 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car2.ID, car2.model, car2.rental_rate_per_day, days, total_rental_cost2);
53
54     printf("\nCar 3 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car3.ID, car3.model, car3.rental_rate_per_day, days, total_rental_cost3);
55
56     printf("\n=== Code Execution Successful ===");
57

```

Enter details for Car 1:  
Car ID: 1  
Model: toyota  
Rental Rate per Day: 2000  
  
Enter details for Car 2:  
Car ID: 2  
Model: swift  
Rental Rate per Day: 1500  
  
Enter details for Car 3:  
Car ID: 3  
Model: honda  
Rental Rate per Day: 3000  
  
Enter the number of days for rental: 7  
  
Car Rental Details:  
  
Car 1 - ID: 1  
Model: toyota  
Rental Rate per Day: 2000.00  
Total Rental Cost for 7 days: 14000.00  
  
Car 2 - ID: 2  
Model: swift  
Rental Rate per Day: 1500.00  
Total Rental Cost for 7 days: 10500.00  
  
Car 3 - ID: 3  
Model: honda  
Rental Rate per Day: 3000.00  
Total Rental Cost for 7 days: 21000.00  
  
=== Code Execution Successful ===

main.c

Share

Run

Output

Clear

```

31
32     printf("\nEnter details for Car 3:\n");
33     printf("Car ID: ");
34     scanf("%d", &car3.ID);
35     getchar();
36     printf("Model: ");
37     fgets(car3.model, sizeof(car3.model), stdin);
38     printf("Rental Rate per Day: ");
39     scanf("%f", &car3.rental_rate_per_day);
40
41     printf("\nEnter the number of days for rental: ");
42     scanf("%d", &days);
43
44     total_rental_cost1 = car1.rental_rate_per_day * days;
45     total_rental_cost2 = car2.rental_rate_per_day * days;
46     total_rental_cost3 = car3.rental_rate_per_day * days;
47
48     printf("\nCar Rental Details:\n");
49
50     printf("\nCar 1 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car1.ID, car1.model, car1.rental_rate_per_day, days, total_rental_cost1);
51
52     printf("\nCar 2 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car2.ID, car2.model, car2.rental_rate_per_day, days, total_rental_cost2);
53
54     printf("\nCar 3 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
        car3.ID, car3.model, car3.rental_rate_per_day, days, total_rental_cost3);
55
56     printf("\n=== Code Execution Successful ===");
57

```

Enter details for Car 2:  
Car ID: 2  
Model: swift  
Rental Rate per Day: 1500  
  
Enter details for Car 3:  
Car ID: 3  
Model: honda  
Rental Rate per Day: 3000  
  
Enter the number of days for rental: 7  
  
Car Rental Details:  
  
Car 1 - ID: 1  
Model: toyota  
Rental Rate per Day: 2000.00  
Total Rental Cost for 7 days: 14000.00  
  
Car 2 - ID: 2  
Model: swift  
Rental Rate per Day: 1500.00  
Total Rental Cost for 7 days: 10500.00  
  
Car 3 - ID: 3  
Model: honda  
Rental Rate per Day: 3000.00  
Total Rental Cost for 7 days: 21000.00  
  
=== Code Execution Successful ===

```
printf("\nCar Rental Details:\n");
```

```
printf("\nCar 1 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
    car1.ID, car1.model, car1.rental_rate_per_day, days, total_rental_cost1);
```

```
printf("\nCar 2 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
    car2.ID, car2.model, car2.rental_rate_per_day, days, total_rental_cost2);
```

```

    printf("\nCar 3 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d
days: %.2f\n",
        car3.ID, car3.model, car3.rental_rate_per_day, days, total_rental_cost3);

    return 0;
}

```

3.Create a structure named Complex to represent a complex number with real and imaginary parts.Write a C program to add and multiply two complex numbers  
#include <stdio.h>

```

struct Complex {
    float real;
    float imaginary;
};

struct Complex add(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real + c2.real;
    result.imaginary = c1.imaginary + c2.imaginary;
    return result;
}

struct Complex multiply(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real * c2.real - c1.imaginary * c2.imaginary;
    result.imaginary = c1.real * c2.imaginary + c1.imaginary * c2.real;
    return result;
}

void display(struct Complex c) {
    if (c.imaginary >= 0)
        printf("%.2f + %.2fi\n", c.real, c.imaginary);
    else
        printf("%.2f - %.2fi\n", c.real, -c.imaginary);
}

int main() {
    struct Complex c1, c2, sum, product;

    // Input complex numbers
    printf("Enter the first complex number (real and imaginary): ");
    scanf("%f %f", &c1.real, &c1.imaginary);

    printf("Enter the second complex number (real and imaginary): ");
    scanf("%f %f", &c2.real, &c2.imaginary);

    sum = add(c1, c2);

```

```
product = multiply(c1, c2);
```

main.c	Output
<pre>1 #include &lt;stdio.h&gt; 2 3 struct Complex { 4     float real; 5     float imaginary; 6 }; 7 8 struct Complex add(struct Complex c1, struct Complex c2) { 9     struct Complex result; 10    result.real = c1.real + c2.real; 11    result.imaginary = c1.imaginary + c2.imaginary; 12    return result; 13 } 14 15 struct Complex multiply(struct Complex c1, struct Complex c2) { 16     struct Complex result; 17    result.real = c1.real * c2.real - c1.imaginary * c2.imaginary; 18    result.imaginary = c1.real * c2.imaginary + c1.imaginary * c2       .real; 19    return result; 20 } 21 22 void display(struct Complex c) { 23     if (c.imaginary &gt;= 0) 24         printf("%.2f + %.2fi\n", c.real, c.imaginary); 25     else 26         printf("%.2f - %.2fi\n", c.real, -c.imaginary); 27 } 28 29 int main() { 30     struct Complex c1, c2, sum, product;</pre>	<pre>Enter the first complex number (real and imaginary): 4.5 3.8 Enter the second complex number (real and imaginary): 2.8 3.9  Sum of the complex numbers: 7.30 + 7.70i Product of the complex numbers: -2.22 + 28.19i  === Code Execution Successful ===</pre>

main.c	Output
<pre>22 void display(struct Complex c) { 23     if (c.imaginary &gt;= 0) 24         printf("%.2f + %.2fi\n", c.real, c.imaginary); 25     else 26         printf("%.2f - %.2fi\n", c.real, -c.imaginary); 27 } 28 29 int main() { 30     struct Complex c1, c2, sum, product; 31 32     printf("Enter the first complex number (real and imaginary): "); 33     scanf("%f %f", &amp;c1.real, &amp;c1.imaginary); 34 35     printf("Enter the second complex number (real and imaginary): "); 36     scanf("%f %f", &amp;c2.real, &amp;c2.imaginary); 37 38     sum = add(c1, c2); 39 40     product = multiply(c1, c2); 41 42     printf("\nSum of the complex numbers: "); 43     display(sum); 44 45     printf("Product of the complex numbers: "); 46     display(product); 47 48     return 0; 49 } 50</pre>	<pre>Enter the first complex number (real and imaginary): 4.5 3.8 Enter the second complex number (real and imaginary): 2.8 3.9  Sum of the complex numbers: 7.30 + 7.70i Product of the complex numbers: -2.22 + 28.19i  === Code Execution Successful ===</pre>

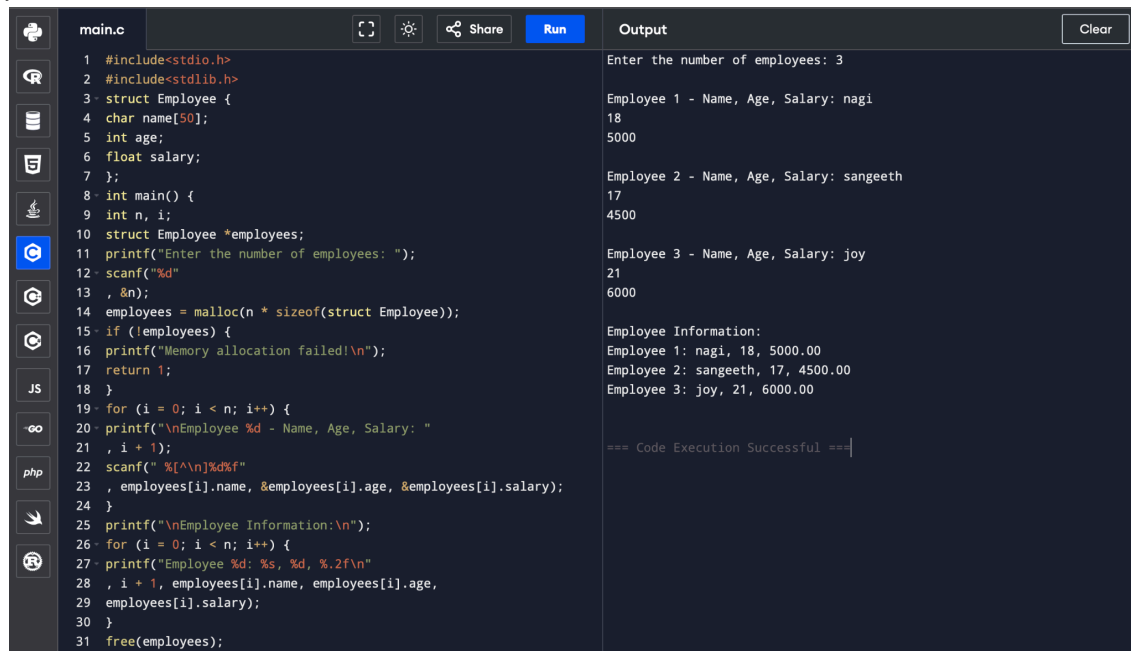
```
printf("\nSum of the complex numbers: ");
display(sum);
```

```
printf("Product of the complex numbers: ");
display(product);
```

```
return 0;
}
```

4. Write a program in C to store and print the information of N employees using dynamic memory allocation and structures.

```
#include<stdio.h>
#include<stdlib.h>
struct Employee {
char name[50];
int age;
float salary;
};
int main() {
int n, i;
struct Employee *employees;
printf("Enter the number of employees: ");
scanf("%d"
, &n);
employees = malloc(n * sizeof(struct Employee));
if (!employees) {
printf("Memory allocation failed!\n");
return 1;
}
for (i = 0; i < n; i++) {
printf("\nEmployee %d - Name, Age, Salary: "
, i + 1);
scanf(" %[^\\n]%d%f"
, employees[i].name, &employees[i].age, &employees[i].salary);
}
printf("\nEmployee Information:\n");
for (i = 0; i < n; i++) {
printf("Employee %d: %s, %d, %.2f\n"
, i + 1, employees[i].name, employees[i].age,
employees[i].salary);
}
free(employees);
return 0;
}
```



main.c	Output
1 #include<stdio.h>	Enter the number of employees: 3
2 #include<stdlib.h>	
3 struct Employee {	Employee 1 - Name, Age, Salary: nagi
4 char name[50];	18
5 int age;	5000
6 float salary;	
7 };	Employee 2 - Name, Age, Salary: sangeeth
8 int main() {	17
9 int n, i;	4500
10 struct Employee *employees;	
11 printf("Enter the number of employees: ");	Employee 3 - Name, Age, Salary: joy
12 scanf("%d"	21
13 , &n);	6000
14 employees = malloc(n * sizeof(struct Employee));	
15 if (!employees) {	Employee Information:
16 printf("Memory allocation failed!\n");	Employee 1: nagi, 18, 5000.00
17 return 1;	Employee 2: sangeeth, 17, 4500.00
18 }	Employee 3: joy, 21, 6000.00
19 for (i = 0; i < n; i++) {	
20 printf("\nEmployee %d - Name, Age, Salary: "	=== Code Execution Successful ===
21 , i + 1);	
22 scanf(" %[^\\n]%d%f"	
23 , employees[i].name, &employees[i].age, &employees[i].salary);	
24 }	
25 printf("\nEmployee Information:\n");	
26 for (i = 0; i < n; i++) {	
27 printf("Employee %d: %s, %d, %.2f\n"	
28 , i + 1, employees[i].name, employees[i].age,	
29 employees[i].salary);	
30 }	
31 free(employees);	