

# 1)

main.c	Run	Output
<pre>1 #include &lt;stdio.h&gt; 2 #include &lt;stdlib.h&gt; 3 struct Person { 4     char *name; 5     int age; 6 }; 7 int main() 8 { 9     struct Person *p; 10    p= ("struct Person"); 11    p-&gt;age = 30; 12    printf("Name: %s\n", p-&gt;name); 13    printf("Age: %d\n", p-&gt;age); 14    free(p); 15    return 0; 16 }</pre>		<pre>/tmp/jHZKi8gHWr.o Segmentation fault</pre>

# 2)

c pro 1.cpp	C:\Users\suris\OneDrive\Desktop\c pro 1.exe
<pre>2 3 struct simp 4 { 5     int i; 6     char city[10]; 7 }; 8 9 int main() 10 { 11     struct simp s1 = {6, "chennai"}; 12 13     printf("%s\n", s1.city); 14     printf("%d\n", s1.i); 15 16     return 0; 17 }</pre>	<pre>chennai 6 ----- Process exited after 0.06276 seconds with return value 0 Press any key to continue . . .</pre>

# 3)

The screenshot shows a C program named 'c pro 1.cpp' and its execution output. The code defines a struct 'zoho' with fields 'employees', 'comp', and 'founder p'. It initializes a 'founder' struct 'f' with 'sridhar' and a 'zoho' struct 'zs' with '4000', 'zoho', and 'f'. It then prints the values of 'zs.comp', 'zs.employees', and 'zs.p.ceo'.

```
c pro 1.cpp
5 {
6     char ceo[10];
7 };
8
9 struct zoho
10 {
11     int employees;
12     char comp[5];
13     struct founder p;
14 };
15
16 struct founder f = {"sridhar"};
17 struct zoho zs = {4000, "zoho", f};
18 printf("%s %d %s", zs.comp, zs.employees, zs.p.ceo);
19 return 0;
20 }
```

The execution output shows the program running successfully, printing 'zoho 4000 sridhar' and exiting after 0.05446 seconds.

```
C:\Users\suris\OneDrive\Desktop\c pro 1.exe
zoho 4000 sridhar
-----
Process exited after 0.05446 seconds with return value 0
Press any key to continue . . .
```

4)

The screenshot shows a C program named 'c pro 4.cpp' and its execution output. The code includes 'stdio.h' and defines a 'main' function that declares an integer 'a' with value 130, a character pointer 'ptr', and assigns 'ptr' to point to 'a'. It then prints the value of 'ptr' and returns 0.

```
c pro 4.cpp
1 #include<stdio.h>
2 int main(){
3     int a = 130;
4     char *ptr;
5     ptr = (char *)&a;
6     printf("%d ",*ptr);
7     return 0;
8 }
```

The execution output shows the program running successfully, printing '130' and exiting after 0.06481 seconds.

```
C:\Users\suris\OneDrive\Desktop\c pro 4.exe
-126
-----
Process exited after 0.06481 seconds with return value 0
Press any key to continue . . .
```

5)

The screenshot shows a C++ IDE with a file named `c pro 5.cpp`. The code defines a structure for an employee and a function to find the employee with the highest salary. The execution window shows the program running, prompting for the number of employees (2), then details for each employee (Employee Number, Name, Salary). It then displays the details of the employee with the highest salary (Employee Number: 192191827, Name: varun, Salary: 60000.00).

```
1 #include <stdio.h>
2 #include <string.h>
3
4 #define MAX_EMPLOYEES 100
5
6 // Define a structure to represent an employee
7 struct employee {
8     int eno;
9     char ename[50];
10    float salary;
11 };
12
13 int main() {
14     int n, i;
15     struct employee employees[MAX_EMPLOYEES];
16     float max_salary = -1.0; // Initialize max_salary
17     int max_salary_employee_index = -1;
```

Execution output:

```
Enter the number of employees: 2
Enter details of employee 1:
Employee Number: 192191018
Employee Name: teja
Employee Salary: 50000
Enter details of employee 2:
Employee Number: 192191827
Employee Name: varun
Employee Salary: 60000

Details of employee with highest salary:
Employee Number: 192191827
Employee Name: varun
Employee Salary: 60000.00

-----
Process exited after 68.91 seconds with return value 0
Press any key to continue . . .
```

6)

The screenshot shows a C++ IDE with a file named `c pro 6.cpp`. The code defines a structure for an employee and a function to create and print an employee's details using a pointer. The execution window shows the program running, displaying the employee's details (Employee ID: 101, Employee Name: John, Employee Salary: 5000.50).

```
10 int main() {
11     struct Employee emp;
12     struct Employee *empPtr = &emp;
13
14
15     empPtr->empID = 101;
16     strcpy(empPtr->empName, "John");
17     empPtr->empSalary = 5000.50;
18
19
20     printf("Employee ID: %d\n", empPtr->empID);
21     printf("Employee Name: %s\n", empPtr->empName);
22     printf("Employee Salary: %.2f\n", empPtr->empSalary);
23
24     return 0;
25 }
```

Execution output:

```
Employee ID: 101
Employee Name: John
Employee Salary: 5000.50

-----
Process exited after 0.06446 seconds with return value 0
Press any key to continue . . .
```

8)

The screenshot shows a C++ IDE with two windows. The left window, titled 'c pro 8.cpp', contains the following code:

```
25     case 'D':
26         countD++;
27         break;
28     case 'F':
29         countF++;
30         break;
31 }
32 }
33 for (int i = 0; i < n; i++) {
34     printf("Student %d details:\n", i);
35     printf("Number: %d\n", student[i].Number);
36     printf("Grade: %c\n", student[i].Grade);
37     printf("Total no. A: %d, B: %d, C: %d, D: %d, F: %d\n",
38           student[i].TotalA, student[i].TotalB, student[i].TotalC,
39           student[i].TotalD, student[i].TotalF);
40 }
```

The right window, titled 'C:\Users\suris\OneDrive\Desktop\c pro 8.exe', shows the program's execution. It prompts the user to enter the number of students (2), then the details for each student. The output shows the details for two students, including their numbers, grades, and the total counts for each grade. The program exits after 8.062 seconds.

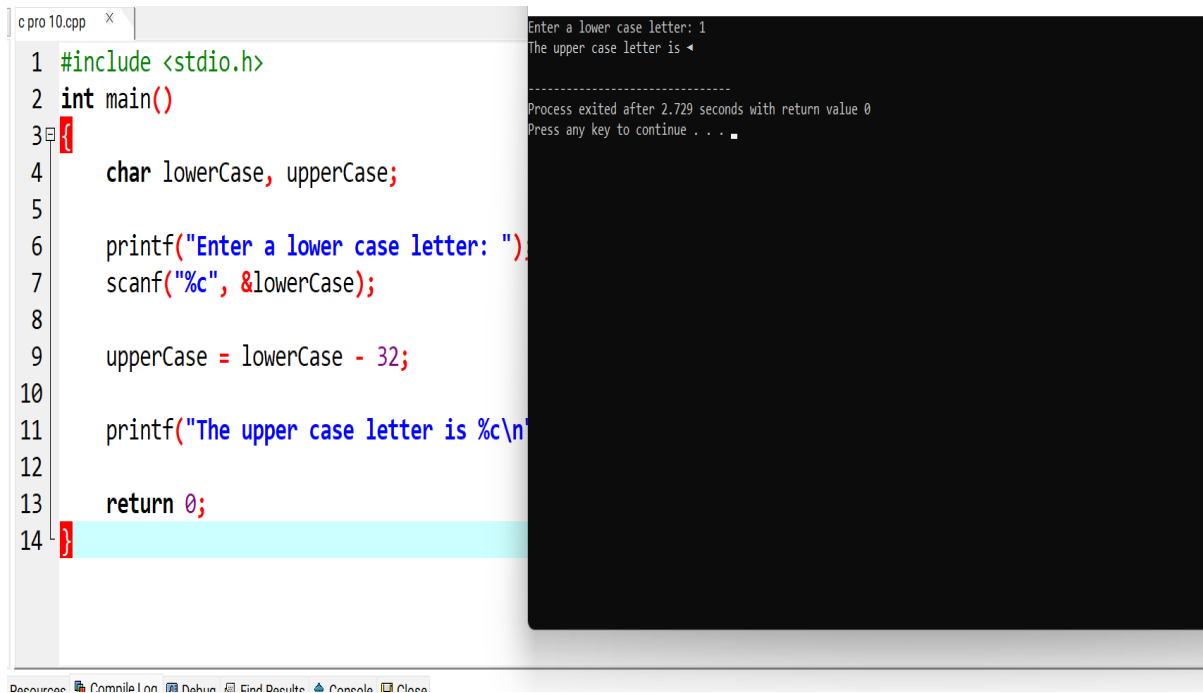
9)

The screenshot shows a C++ IDE with two windows. The left window, titled 'c pro 9.cpp', contains the following code:

```
3     int i, hcf;
4     for (i = 1; i <= num1 && i <= num2; i++)
5     {
6         if (num1 % i == 0 && num2 % i == 0)
7             hcf = i;
8     }
9     return hcf;
10 }
11 int main() {
12     int num1, num2, hcf;
13     printf("Enter two numbers: ");
14     scanf("%d %d", &num1, &num2);
15     hcf = findHCF(num1, num2);
16     printf("HCF or GCD of %d and %d is %d\n", num1, num2, hcf);
17     return 0;
18 }
```

The right window, titled 'C:\Users\suris\OneDrive\Desktop\c pro 9.exe', shows the program's execution. It prompts the user to enter two numbers (2 and 3). The output shows the HCF or GCD of 2 and 3 is 1. The program exits after 3.604 seconds.

# 10)



The image shows a C++ program in a code editor on the left and its execution output in a terminal window on the right. The code is a simple program that takes a lowercase letter as input and prints its corresponding uppercase letter. The terminal shows the program running with the input 'l' and output 'L'.

```
c pro 10.cpp x
1 #include <stdio.h>
2 int main()
3 {
4     char lowerCase, upperCase;
5
6     printf("Enter a lower case letter: ");
7     scanf("%c", &lowerCase);
8
9     upperCase = lowerCase - 32;
10
11     printf("The upper case letter is %c\n", upperCase);
12
13     return 0;
14 }
```

```
Enter a lower case letter: l
The upper case letter is L

-----
Process exited after 2.729 seconds with return value 0
Press any key to continue . . .
```