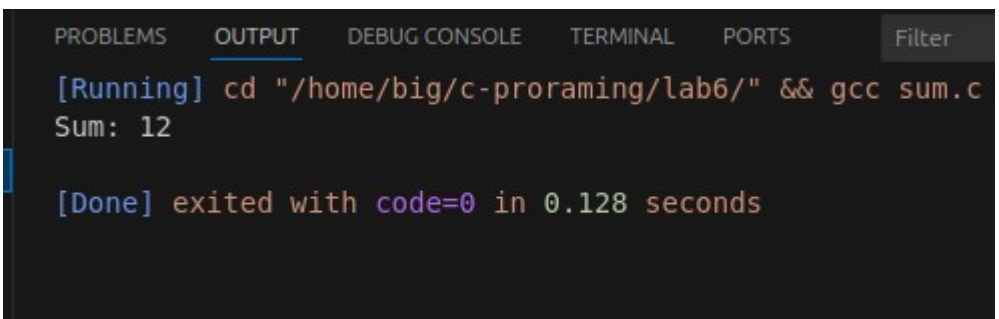
**Lab VI (Functions)** *-Kishor Neupane*

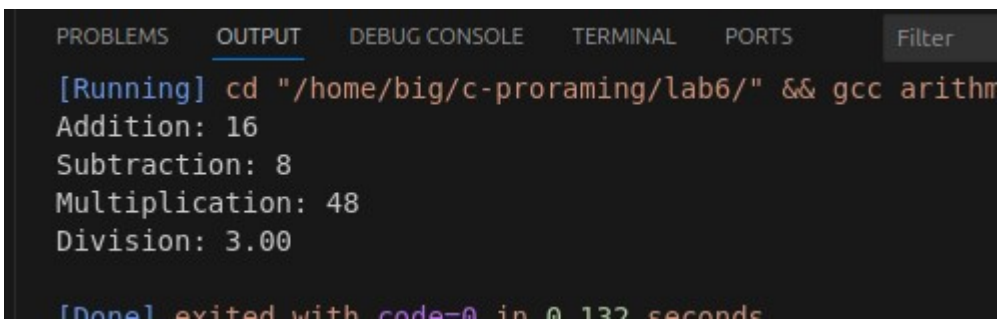
1. Write a function sum that takes two integers as arguments and returns their sum. Call this function from the main function and print the result.

**Output :**



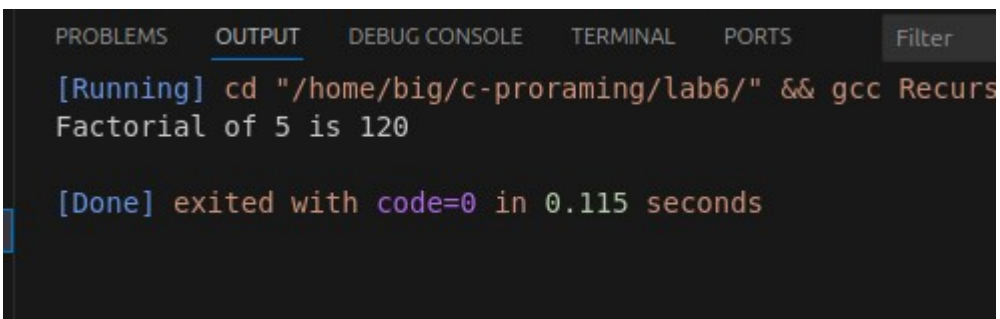
2. Write a program that demonstrates nesting of functions, where each function performs a specific arithmetic operation (addition, subtraction, multiplication, division) on two numbers.

**Output :**



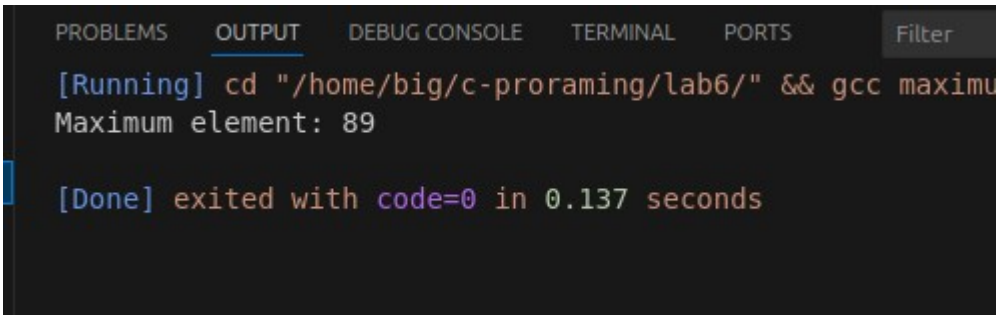
3. Write a recursive function factorial to calculate the factorial of a given integer. Use this function to find the factorial of 5.

**Output :**



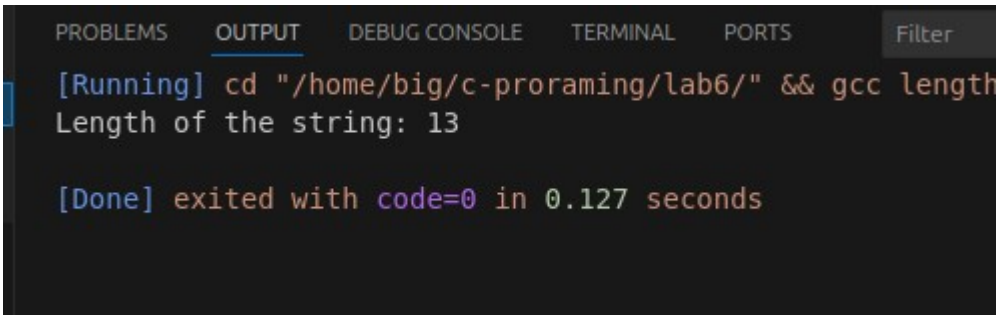
4. Write a function findMax that takes an array of integers and its size as arguments and returns the maximum element in the array.

**Output :**



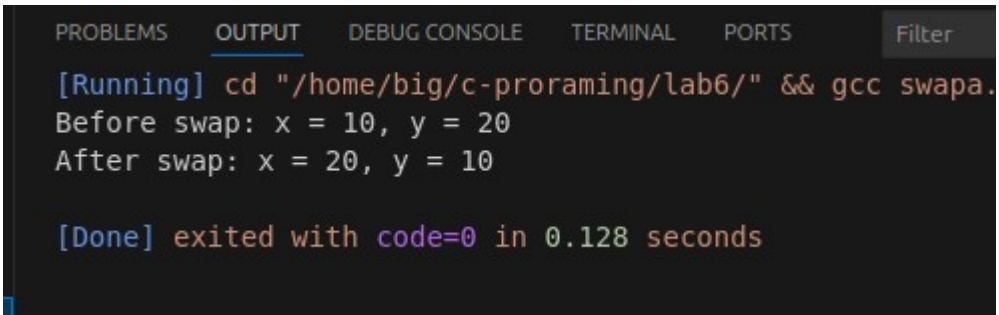
5. Write a function stringLength that takes a string as an argument and returns its length.

**Output :**



6. Write a program that demonstrates passing arguments by address. Define a function swap that takes two integer pointers as arguments and swaps the values they point to. Print the swapped values from the main function.

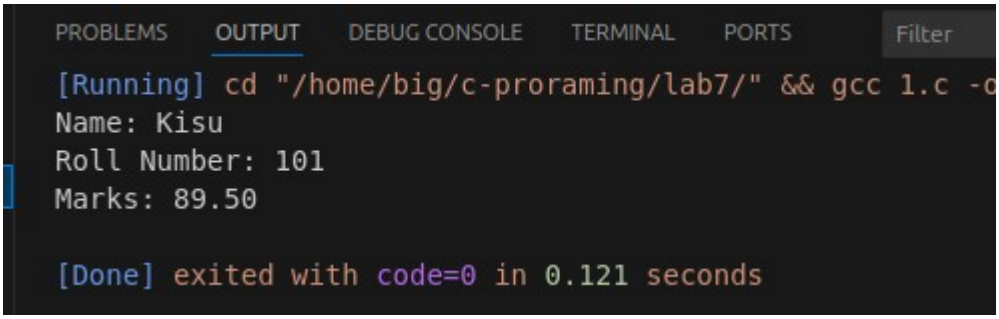
**Output :**



**Lab VII (Structure and Union)** *-Kishor Neupane*

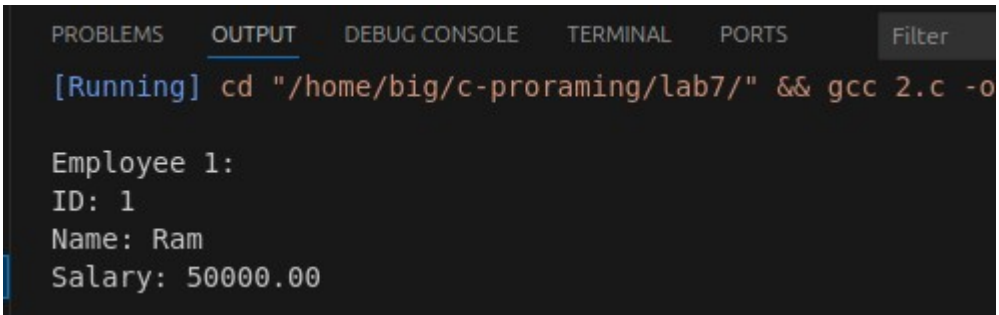
1. Define a structure called Student with the following members: name, roll_number, and marks. Write a program to create a variable of type Student, initialize its members, and print them.

**Output :**



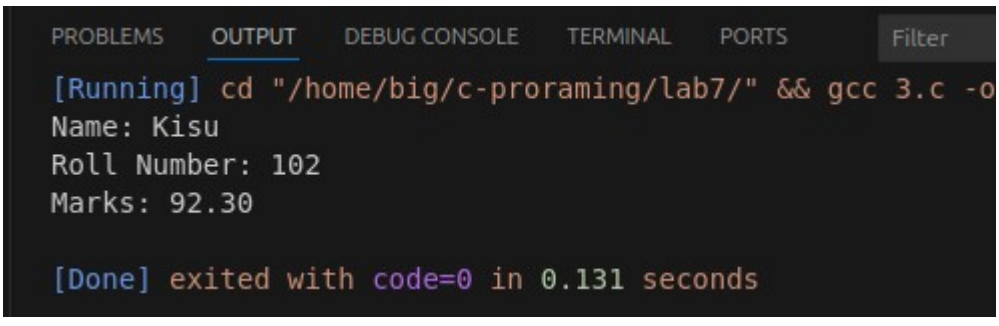
2. Define a structure called Employee with members emp_id, name, and salary. Write a program to create an array of 5 Employee structures, initialize them, and print the details of each employee.

**Output :**



3. Define a function displayStudent that takes a Student structure as a parameter and prints its details.

**Output :**

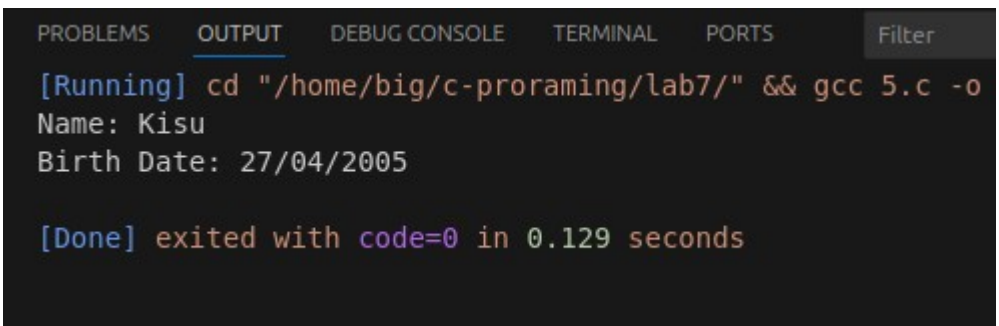


4. Define a function displayEmployees that takes an array of Employee structures and its size as parameters, and prints the details of all employees.

**Output :**

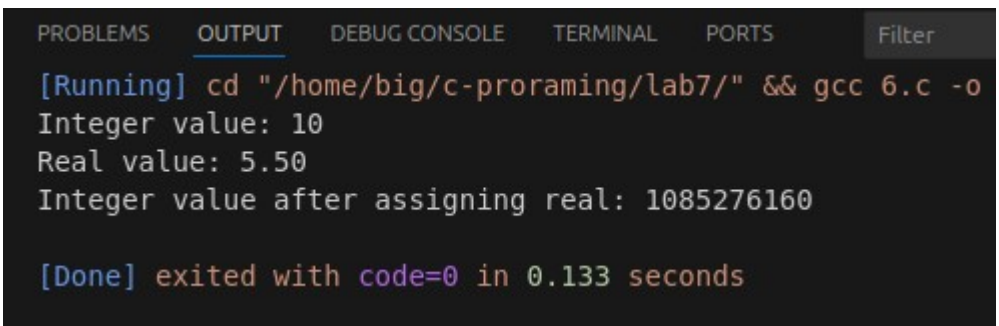
5. Define a structure called Date with members day, month, and year. Define another structure called Person with members name and birth_date of type Date. Write a program to create a Person structure and print its details.

**Output :**



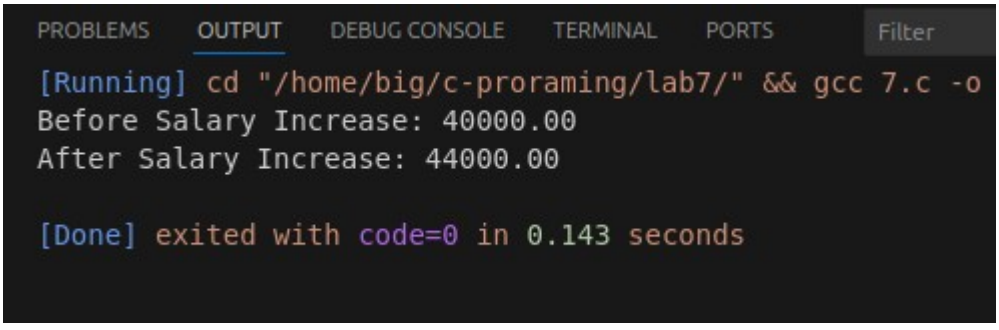
6. Define a union called Number with members integer and real of types int and float respectively. Write a program to demonstrate the use of this union by storing an integer and a real number in it, and printing them.

**Output :**



7. Define a function changeSalary that takes a pointer to an Employee structure and increases its salary by 10%. Use this function to update the salary of an employee.

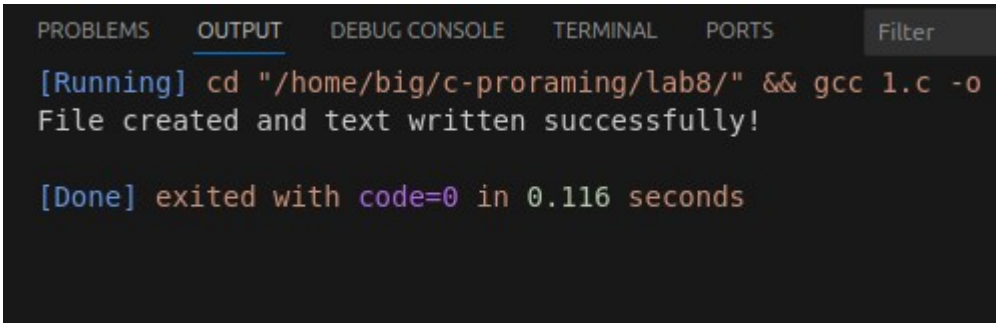
**Output :**



**Lab VIII (File Handling)** *-Kishor Neupane*

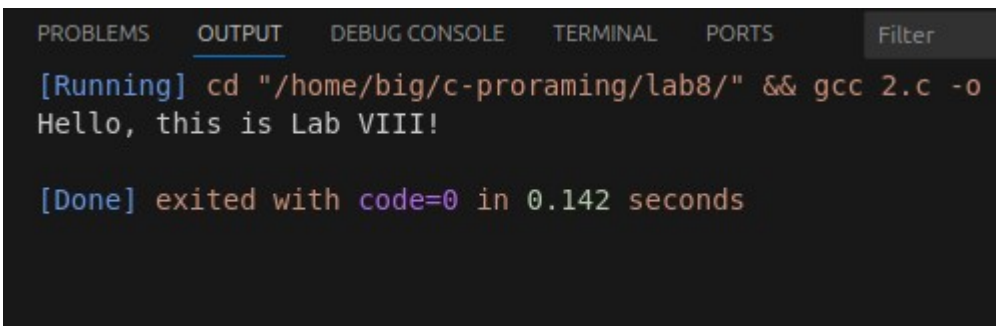
1. Write a program in C to create a file and write some text into it. Close the file and display a message indicating successful completion.

**Output :**



2. Write a program in C to open an existing file and display its contents on the screen.

**Output :**



3. Write a program in C to append data to an existing file. Display the contents of the file before and after appending.

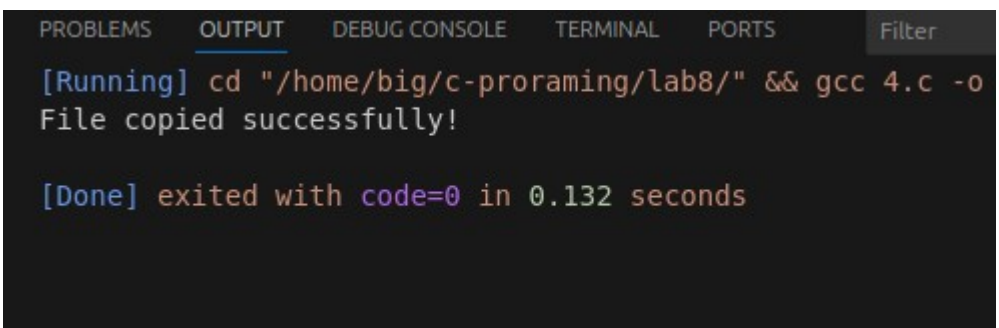
**Output :**

```
[Running] cd "/home/big/c-proraming/lab8/" && gcc 3.c -o
Before appending:
Hello, this is Lab VIII!

After appending:
Hello, this is Lab VIII!
Appending new text!
```

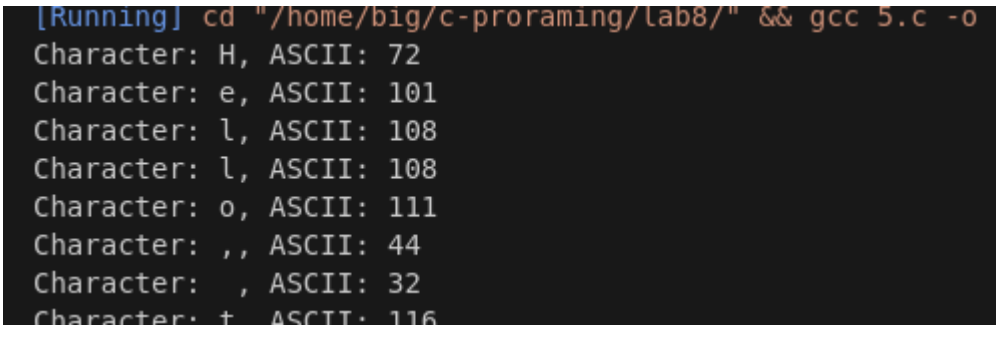
4. Write a program in C to copy the contents of one file into another file.

**Output :**



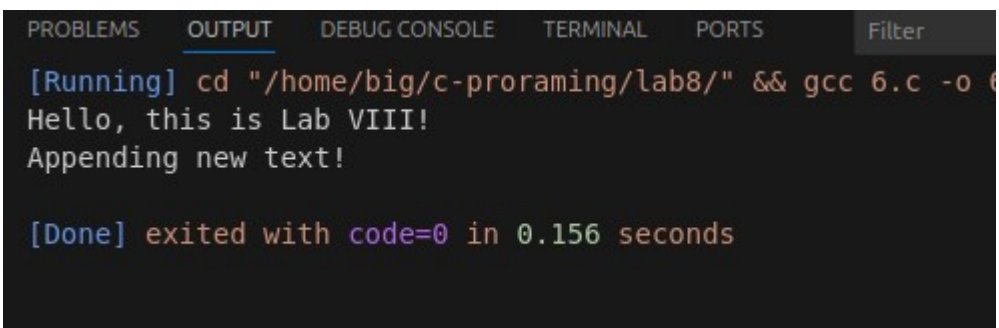
5. Write a program in C to read a file character by character and display the characters along with their ASCII values.

**Output :**



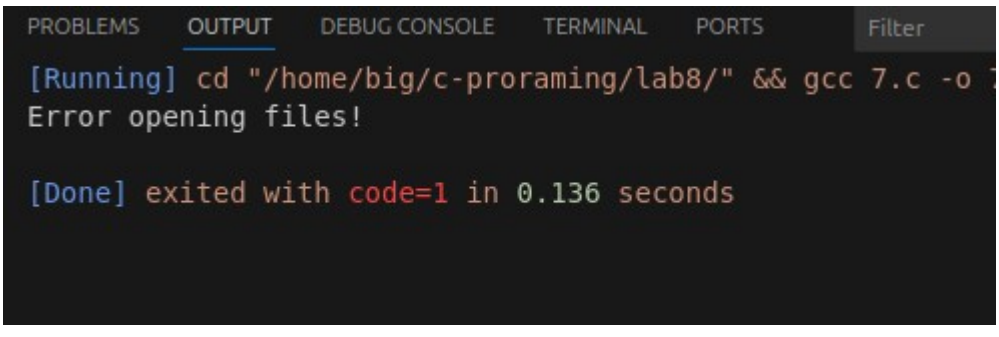
6. Write a program in C to read a file line by line and display each line on the screen.

**Output :**



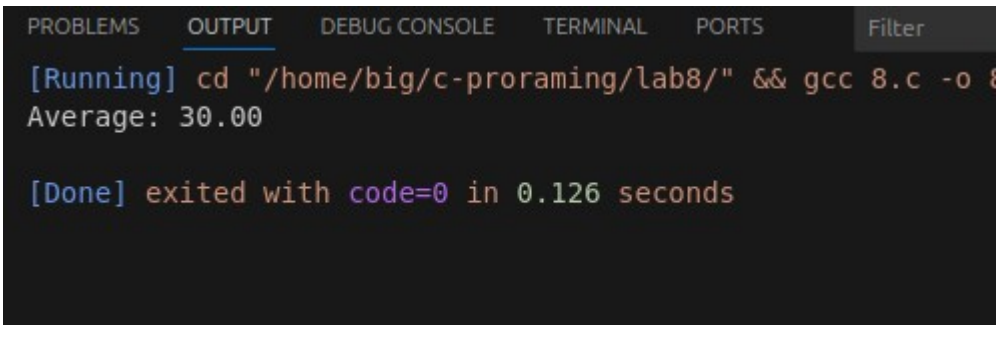
7. Write a program in C to read an integer from a file and calculate its square. Write the square to another file.

**Output :**



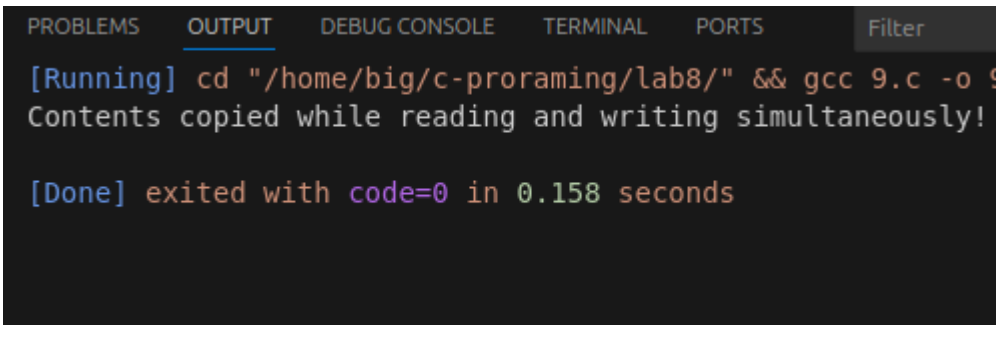
8. Write a program in C to create a file with random numbers and then read the numbers from the file and calculate their average.

**Output :**



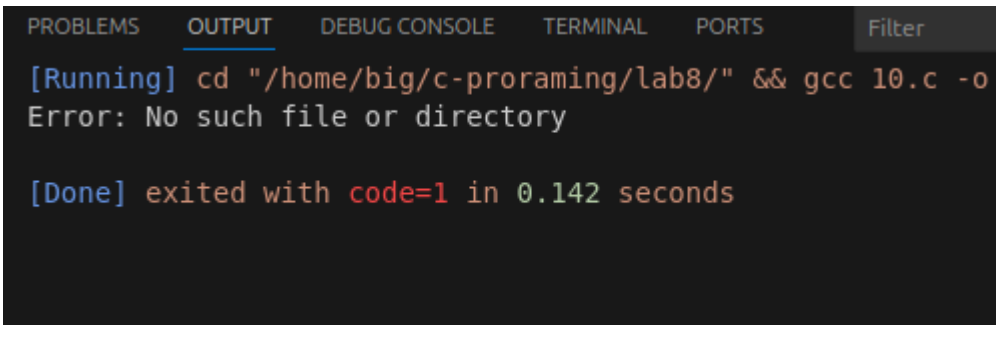
9. Write a program in C to open a file in read mode and write mode simultaneously. Read from one file and write to the other file simultaneously.

**Output :**



10. Write a program in C to demonstrate error handling in file operations. Handle errors such as file not found, permission denied, etc., and display appropriate messages.

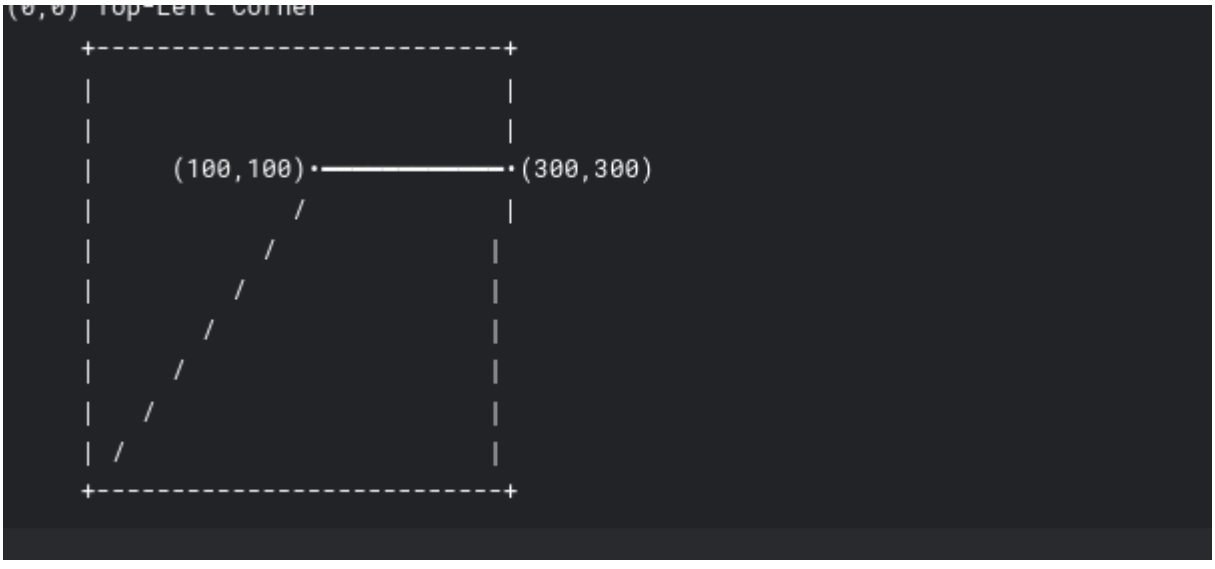
**Output :**



**Lab IX (Graphics)** *-Kishor Neupane*

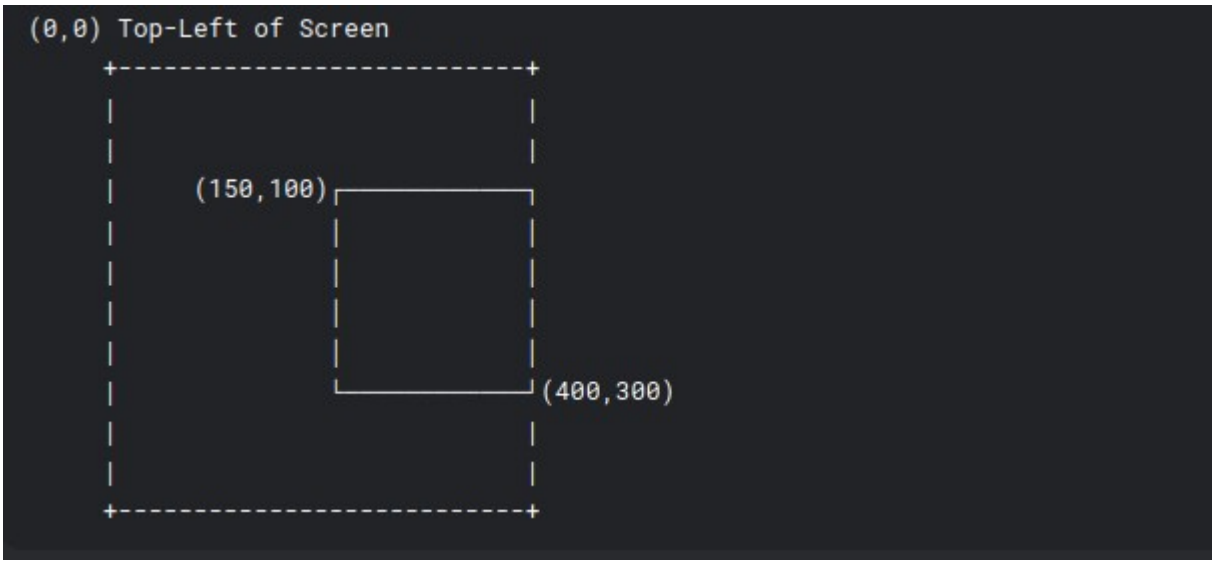
1. Write a program in C to draw a line.

**Output :**



2. Write a program to draw a rectangle.

**Output :**



3. Write a program to draw a circle in C.

**Output :**