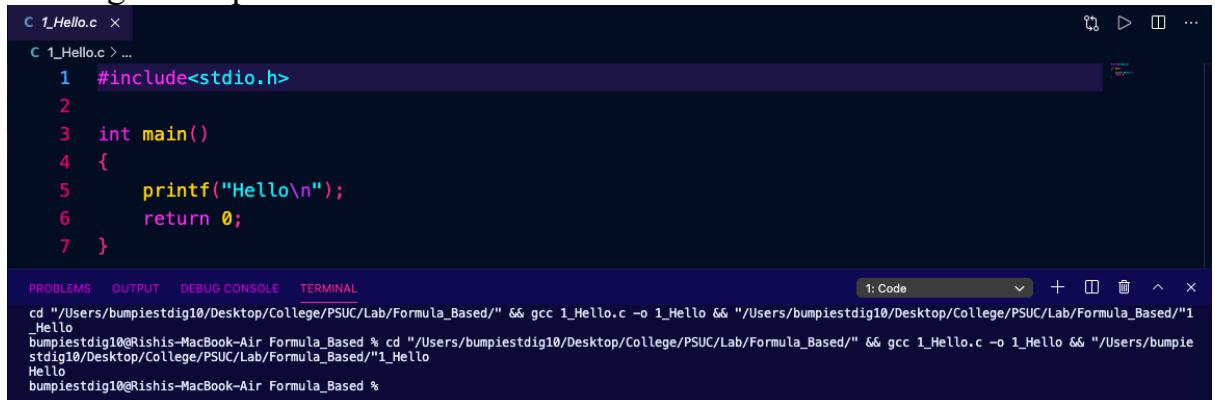


# PROBLEM SOLVING USING COMPUTERS

## Lab 1. Sequential Instruction Based C Programs

1. Program to print “Hello” on the Screen.



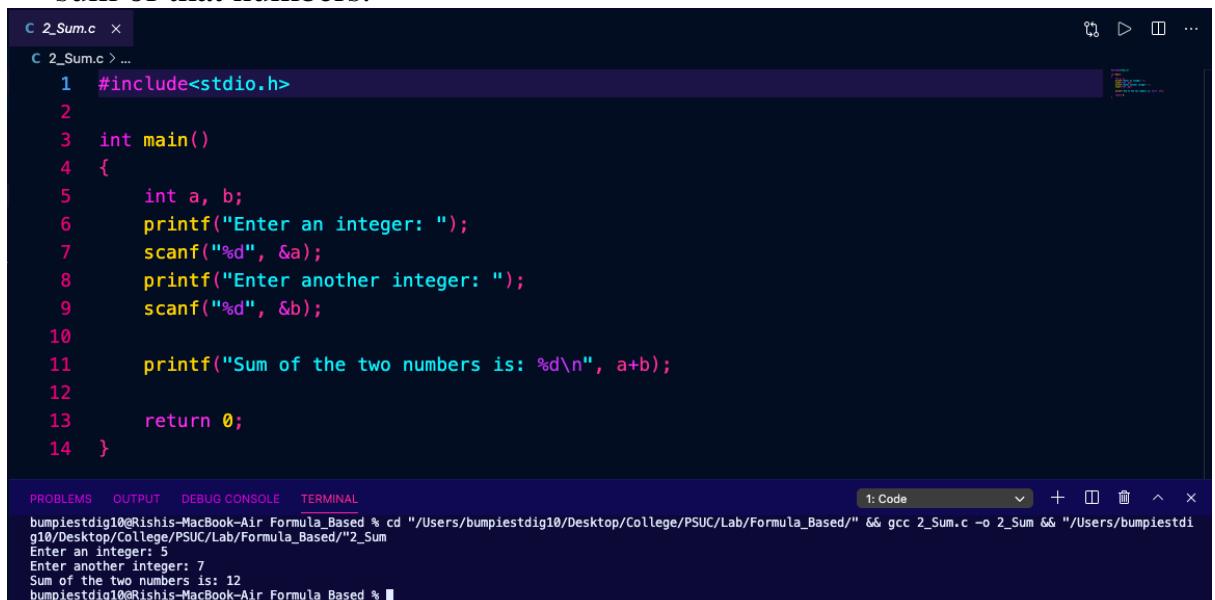
```

C 1_Hello.c ✘
C 1_Hello.c > ...
1 #include<stdio.h>
2
3 int main()
4 {
5     printf("Hello\n");
6     return 0;
7 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ^ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 1_Hello.c -o 1_Hello && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 1_Hello
Hello
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 1_Hello.c -o 1_Hello && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 1_Hello
Hello
bumpiestdig10@Rishis-MacBook-Air Formula_Based %

```

2. Write a program to take an input of two integer numbers and print the sum of that numbers.



```

C 2_Sum.c ✘
C 2_Sum.c > ...
1 #include<stdio.h>
2
3 int main()
4 {
5     int a, b;
6     printf("Enter an integer: ");
7     scanf("%d", &a);
8     printf("Enter another integer: ");
9     scanf("%d", &b);
10
11    printf("Sum of the two numbers is: %d\n", a+b);
12
13    return 0;
14 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ^ ×
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 2_Sum.c -o 2_Sum && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 2_Sum
Enter an integer: 5
Enter another integer: 7
Sum of the two numbers is: 12
bumpiestdig10@Rishis-MacBook-Air Formula_Based %

```

## PROBLEM SOLVING USING COMPUTERS LAB

3. Convert the time in seconds to hours, minutes and seconds. (1 hr =3600 sec).

The screenshot shows a code editor window with the following C code:

```
C 3_Time.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int seconds;
5     int hours;
6     int minutes;
7
8     printf("Enter seconds: ");
9     scanf("%d", &seconds);
10
11    hours = seconds/3600;
12    minutes = (seconds%3600)/60;
13    seconds = ((seconds%3600)%60);
14
15    printf("%dhour(s) %dminute(s) %dsecond(s)\n", hours,minutes,seconds);
16
17    return 0;
18 }
```

Below the code, the terminal output shows the program running and outputting "1hour(s) 1minute(s) 15second(s)".

4. Find the sum of the digits of a four-digit number (ex 1234 sum=10) (without using a loop).

The screenshot shows a code editor window with the following C code:

```
C 4_Four_Digit_Number.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int number;
5     int temp;
6     int sum = 0;
7     printf("Enter a four digit number: ");
8     scanf("%d", &number);
9
10    temp = number;
11    sum = sum + (temp%10);
12    temp = temp/10;
13    sum = sum + (temp%10);
14    temp = temp/10;
15    sum = sum + (temp%10);
16    temp = temp/10;
17    sum = sum + (temp%10);
18
19    printf("Sum of the digits: %d\n", sum);
20
21    return 0;
22 }
```

Below the code, the terminal output shows the program running and outputting "Sum of the digits: 10".

5. Convert temperature given in Fahrenheit to Centigrade and Centigrade to Fahrenheit. Hint:  $C=5/9(F-32)$ .

```

5_Temp.c x
C 5_Temp.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     float c=0, f=0;
6     int option;
7
8     printf("Entering temperature in: (1)Centigrade (2)Fahrenheit\n");
9     scanf("%d", &option);
10
11    if(option == 1)
12    {
13        printf("Temperature: ");
14        scanf("%f", &c);
15        f = ((9*c)/5)+32;
16        printf("In Fahrenheit: %.2f\n", f);
17    }
18    else if(option == 2)
19    {
20        printf("Temperature: ");
21        scanf("%f", &f);
22        c = (5*(f-32))/9;
23        printf("In Centigrade: %.2fc\n", c);
24    }
25    else
26    {
27        printf("Invalid Input!\n");
28    }
29
30    return 0;
31 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ■ ^ X

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 5_Temp.c -o 5_Temp && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"5_Temp
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 5_Temp.c -o 5_Temp && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"5_Temp
Entering temperature in: (1)Centigrade (2)Fahrenheit
1
Temperature: 33.5
In Fahrenheit: 92.30f
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 5_Temp.c -o 5_Temp && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"5_Temp
Entering temperature in: (1)Centigrade (2)Fahrenheit
2
Temperature: 98.6
In Centigrade: 37.00c
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 5_Temp.c -o 5_Temp && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"5_Temp
Entering temperature in: (1)Centigrade (2)Fahrenheit
3
Invalid Input!
bumpiestdig10@Rishis-MacBook-Air Formula_Based %

```

# PROBLEM SOLVING USING COMPUTERS LAB

6. Converting distance in mm to cm, inch, feet (1 cm =10mm, 1inch=2.5cm, 1 feet =12 inches).



```
C 6_Distance.c ×
C 6_Distance.c > ⊜ main()
1 #include<stdio.h>
2 int main()
3 {
4     float mm, cm, inch, feet;
5
6     printf("Enter distance in mm: ");
7     scanf("%f", &mm);
8
9     cm = mm/10.0;
10    inch = cm/2.5;
11    feet = inch/12.0;
12
13 | printf("%.2fmm = %.2fcm = %.2finch = %.2ffeet\n", mm,cm,inch,feet);
14
15    return 0;
16 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 6_Distance.c -o 6_Distance
bumpiestdig10@Rishis-MacBook-Air:~/Desktop/College/PSUC/Lab/Formula_Based% cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 6_Distance.c -o 6_Distance && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 6_Distance
Enter distance in mm: 345.45
345.45mm = 34.55cm = 13.82inch = 1.15feet
bumpiestdig10@Rishis-MacBook-Air:~/Desktop/College/PSUC/Lab/Formula_Based%
```

7. Find out the distance between two points e.g. (x1, y1) and (x2, y2).  
Hint: Distance= $\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$

Hint: Distance= $\sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$

The screenshot shows a code editor window with the following details:

- Title Bar:** The title bar displays "C 7\_Coordinates.c x".
- Code Area:** The main area contains C code for calculating the distance between two points. The code includes prompts for input coordinates, calculations using the distance formula, and a printf statement to output the result.
- Terminal Output:** Below the code editor, the terminal output shows the execution of the program, including user input and the calculated distance.
- Bottom Navigation:** The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, along with other standard UI elements like a search bar and file icons.

```
C 7_Coordinates.c x
C 7_Coordinates.c > ⌂ main()
1 #include<stdio.h>
2 #include<math.h>
3
4 int main()
5 {
6     float x1, y1, x2, y2;
7     double distance;
8
9     printf("Enter the coordinates-\n");
10    printf("x1: ");
11    scanf("%f", &x1);
12    printf("y1: ");
13    scanf("%f", &y1);
14    printf("x2: ");
15    scanf("%f", &x2);
16    printf("y2: ");
17    scanf("%f", &y2);
18
19    distance = sqrt(((x2-x1)*(x2-x1)) + ((y2-y1)*(y2-y1)));
20
21    printf["Distance between (%.2f,% .2f) and (%.2f,% .2f)= %.2f\n",x1,y1,x2,y2, distance];
22
23    return 0;
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 7_Coordinates.c -o 7_Coordinates && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 7_Coordinates
ula_Based/"7_Coordinates
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 7_Coordinates.c -o 7_Coordinates && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 7_Coordinates
Enter the coordinates-
x1: 4.23
y1: 5.43
x2: 1.56
y2: 3.22
Distance between (4.23,5.43) and (1.56,3.22)= 3.47
bumpiestdig10@Rishis-MacBook-Air Formula_Based %
```

8. Evaluate the area of the circle Area =  $\pi * R^2$ 

```
c 8_Circle.c ×
C 8_Circle.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     float radius, area;
6
7     printf("Enter radius of the circle: ");
8     scanf("%f", &radius);
9
10    area = 22*radius*radius/7.0;
11    printf("Area = %.2f\n", area);
12
13    return 0;
14 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ■ ^ ×

```
bumpiestdig10@Rishi-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 8_Circle.c -o 8_Circle && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"8_Circle
Enter radius of the circle: 5.4
Area = 91.65
bumpiestdig10@Rishi-MacBook-Air Formula_Based %
```

## 9. Interchange values of two variables using a third variable.

```
c 9_Interchange3.c ×
C 9_Interchange3.c > main()
1 #include<stdio.h>
2
3 int main()
4 [
5     int var1, var2, var3;
6     printf("Enter variable 1: ");
7     scanf("%d", &var1);
8     printf("Enter variable 2: ");
9     scanf("%d", &var2);
10
11    var3 = var2;
12    printf("\nAfter interchanging, variable 1 = %d ", var3);
13    var3 = var1;
14    printf("and variable 2 = %d\n", var3);
15
16    return 0;
17 ]
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ■ ^ ×

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 9_Interchange3.c -o 9_Interchange3 && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"9_Interchange3 && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/"9_Interchange3
Enter variable 1: 5
Enter variable 2: 7
After interchanging, variable 1 = 7 and variable 2 = 5
bumpiestdig10@Rishi-MacBook-Air Formula_Based %
```

10. Interchange values of two variables without using a third variable.

The screenshot shows a code editor interface with a dark theme. The top bar has tabs for '10\_Interchange2.c' and 'main()' with a save icon. Below the tabs is a toolbar with icons for file operations. The main area contains the following C code:

```
C 10_Interchange2.c ×
C 10_Interchange2.c > ⊖ main()
1 #include<stdio.h>
2
3 int main()
4 {
5     int var1, var2;
6
7     printf("Enter variable 1: ");
8     scanf("%d", &var1);
9     printf("Enter variable 2: ");
10    scanf("%d", &var2);
11
12    printf("After interchanging, variable 1 = %d and variable 2 = %d\n", var2, var1);
13
14    return 0;
15 }
```

Below the code editor is a terminal window titled 'TERMINAL'. It shows the command to run the program, the input of two integers (5 and 7), and the output showing the values swapped.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ×
bumpiestdig10@Rishis-MacBook-Air Formula_Based % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" && gcc 10_Interchange2.c -o 10_Interchange2 & "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Formula_Based/" 10_Interchange2
Enter variable 1: 5
Enter variable 2: 7
After interchanging, variable 1 = 7 and variable 2 = 5
bumpiestdig10@Rishis-MacBook-Air Formula_Based %
```

## Lab 2. Control Structures: If, If-Else and Switch-Case statements

1. Check whether the given number is odd or even.

The screenshot shows a code editor interface with a dark theme. The code file is named `1_Even_Odd.c`. The code itself is as follows:

```

1 #include <stdio.h>
2 int main()
3 {
4     int num;
5     printf("Enter an integer: ");
6     scanf("%d", &num);
7
8     if(num%2 == 0)
9     {
10        printf("%d is an even number.\n", num);
11    }
12    else
13    {
14        printf("%d is an odd number.\n", num);
15    }
16
17    return 0;
18 }

```

Below the code editor is a terminal window showing the execution of the program. The user enters the number `-54`, and the program outputs `-54 is an even number.`. The user then enters `23`, and the program outputs `23 is an odd number.`

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ^ X
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 1_Even_Odd.c -o 1_Even_Odd && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"1_Even_Odd
Enter an integer: -54
-54 is an even number.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 1_Even_Odd.c -o 1_Even_Odd && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"1_Even_Odd
Enter an integer: 23
23 is an odd number.
bumpiestdig10@Rishis-MacBook-Air If_Statement %

```

## PROBLEM SOLVING USING COMPUTERS LAB

### 2. Check whether a given year is a leap year or not.

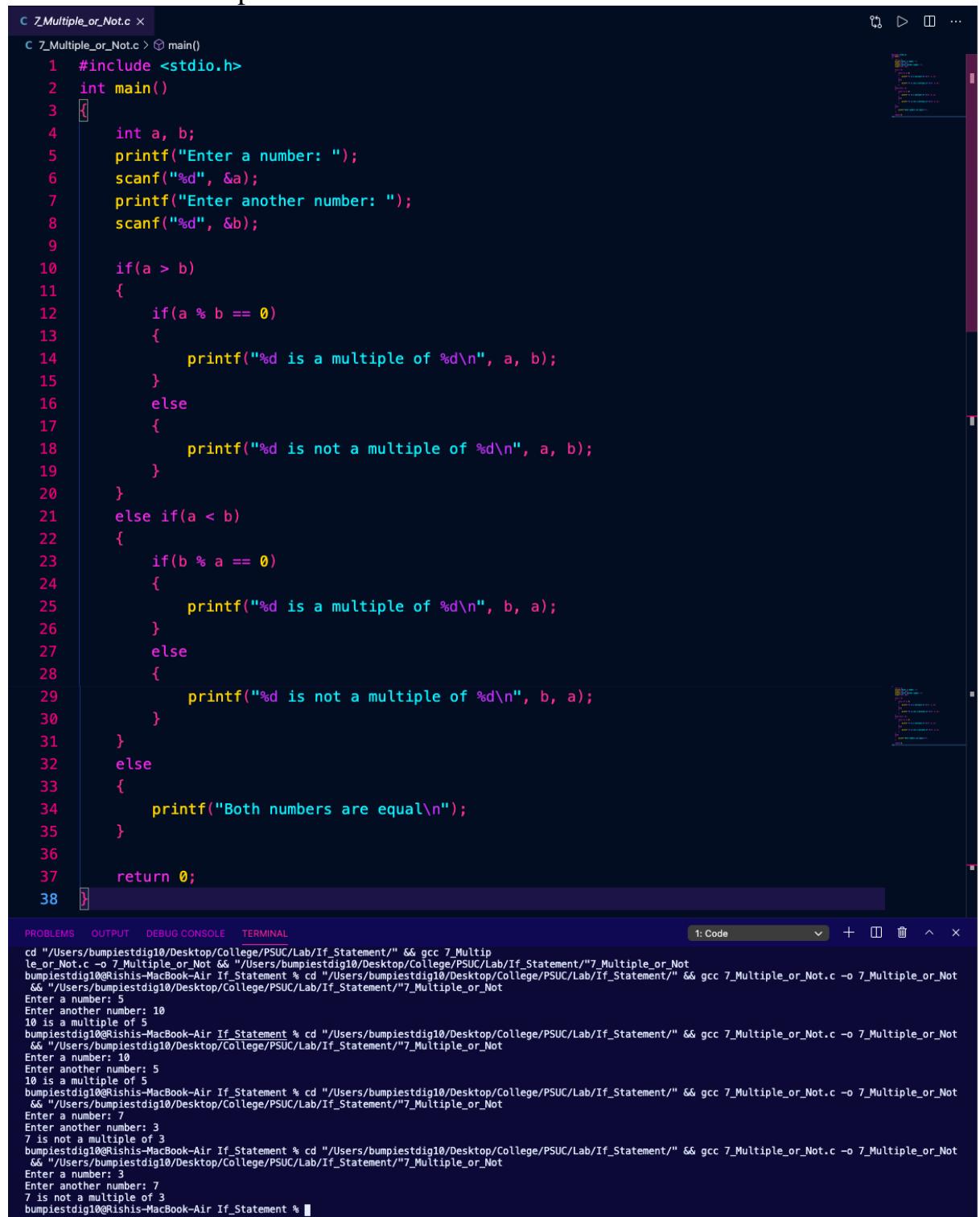
The screenshot shows a code editor window with a dark theme. The file is named `3_Leap_Year.c`. The code implements a function to determine if a given year is a leap year based on the rules: a year is a leap year if it is divisible by 4, but not by 100 unless it is also divisible by 400. The code uses nested if statements to check these conditions and prints the result. Below the code editor is a terminal window showing the execution of the program and its output for the years 2020 and 2021.

```
1 #include <stdio.h>
2 int main()
3 {
4     int year;
5     printf("Enter a year: ");
6     scanf("%d", &year);
7
8     if(year%4 == 0)
9     {
10         if(year%100 == 0)
11         {
12             if(year%400 == 0)
13             {
14                 printf("It is a leap year.\n");
15             }
16             else
17             {
18                 printf("It is not a leap year.\n");
19             }
20         }
21         else
22         {
23             printf("It is a leap year.\n");
24         }
25     }
26     else
27     {
28         printf("It is not a leap year.\n");
29     }
30 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 3_Leap_Year.c -o 3_Leap_Year && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 3_Leap_Year.c -o 3_Leap_Year && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"3_Leap_Year
Enter a year: 2020
It is a leap year.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 3_Leap_Year.c -o 3_Leap_Year && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"3_Leap_Year
Enter a year: 2021
It is not a leap year.
bumpiestdig10@Rishis-MacBook-Air If_Statement %
```

3. Write a program to take two numbers as an input and find whether one number is multiple of other or not.



```

C 7_Multiple_or_Not.c ×
C 7_Multiple_or_Not.c > ⌂ main()
1 #include <stdio.h>
2 int main()
3 {
4     int a, b;
5     printf("Enter a number: ");
6     scanf("%d", &a);
7     printf("Enter another number: ");
8     scanf("%d", &b);
9
10    if(a > b)
11    {
12        if(a % b == 0)
13        {
14            printf("%d is a multiple of %d\n", a, b);
15        }
16        else
17        {
18            printf("%d is not a multiple of %d\n", a, b);
19        }
20    }
21    else if(a < b)
22    {
23        if(b % a == 0)
24        {
25            printf("%d is a multiple of %d\n", b, a);
26        }
27        else
28        {
29            printf("%d is not a multiple of %d\n", b, a);
30        }
31    }
32    else
33    {
34        printf("Both numbers are equal\n");
35    }
36
37    return 0;
38 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 7_Multip
le_or_Not.c -o 7_Multiple_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"7_Multiple_or_N
ot bumpiestdig10@Rishi's-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 7_Multiple_or_Not.c -o 7_Multiple_or_N
ot && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"7_Multiple_or_N
ot Enter a number: 5
Enter another number: 10
10 is a multiple of 5
bumpiestdig10@Rishi's-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 7_Multiple_or_Not.c -o 7_Multiple_or_N
ot && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"7_Multiple_or_N
ot Enter a number: 10
Enter another number: 5
10 is a multiple of 5
bumpiestdig10@Rishi's-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 7_Multiple_or_Not.c -o 7_Multiple_or_N
ot && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"7_Multiple_or_N
ot Enter a number: 3
Enter another number: 7
7 is not a multiple of 3
bumpiestdig10@Rishi's-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 7_Multiple_or_Not.c -o 7_Multiple_or_N
ot && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/"7_Multiple_or_N
ot Enter a number: 3
Enter another number: 7
7 is not a multiple of 3
bumpiestdig10@Rishi's-MacBook-Air If_Statement %

```

4. Write a program that returns a letter grade based on a quiz score. The input will be the integer score from a ten-point quiz.

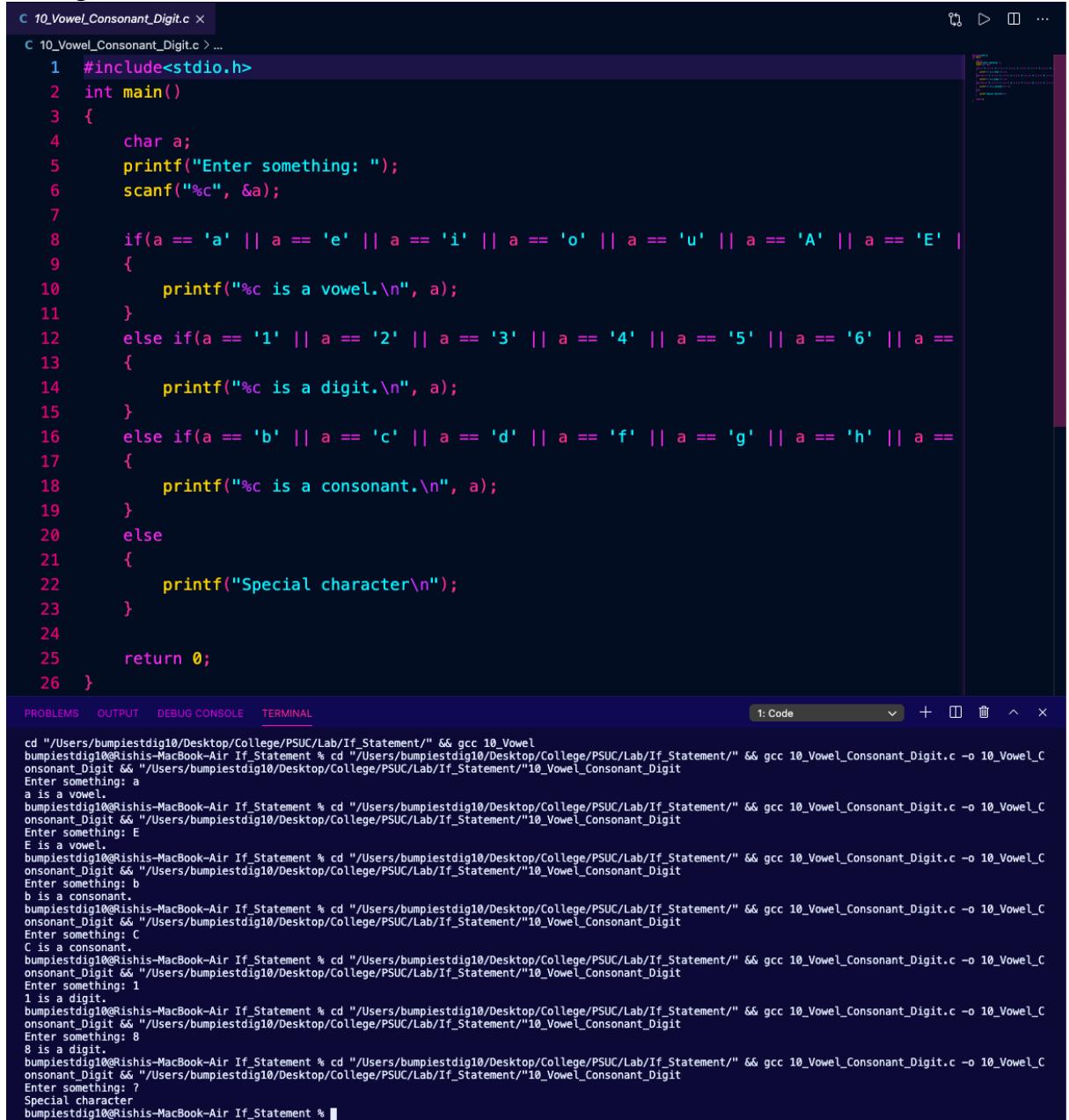
- The letter grades are assigned by:
- 9 – 10 “A” 7 – 8 “B” 5 – 6 “C” 3 – 4 “D” < 3 “F”

```
c 8_Grade.c ×
c 8_Grade.c > ...
1 #include<stdio.h>
2 int main()
3 {
4     int score;
5     printf("Enter quiz score: ");
6     scanf("%d", &score);
7
8     if(score >= 0 && score <= 10)
9     {
10         if(score > 8)
11         {
12             printf("Grade: A\n");
13         }
14         else if(score >= 7 && score <= 8)
15         {
16             printf("Grade: B\n");
17         }
18         else if(score >= 5 && score <= 6)
19         {
20             printf("Grade: C\n");
21         }
22         else if(score >= 3 && score <= 4)
23         {
24             printf("Grade: D\n");
25         }
26         else
27         {
28             printf("Grade: F\n");
29         }
30     }
31     else
32     {
33         printf("Invalid score.\n");
34     }
35
36     return 0;
37 }
```

PROBLEM SOLVING USING COMPUTERS LAB

## PROBLEM SOLVING USING COMPUTERS LAB

5. Write a program to check whether given character is vowel, consonant or digit.



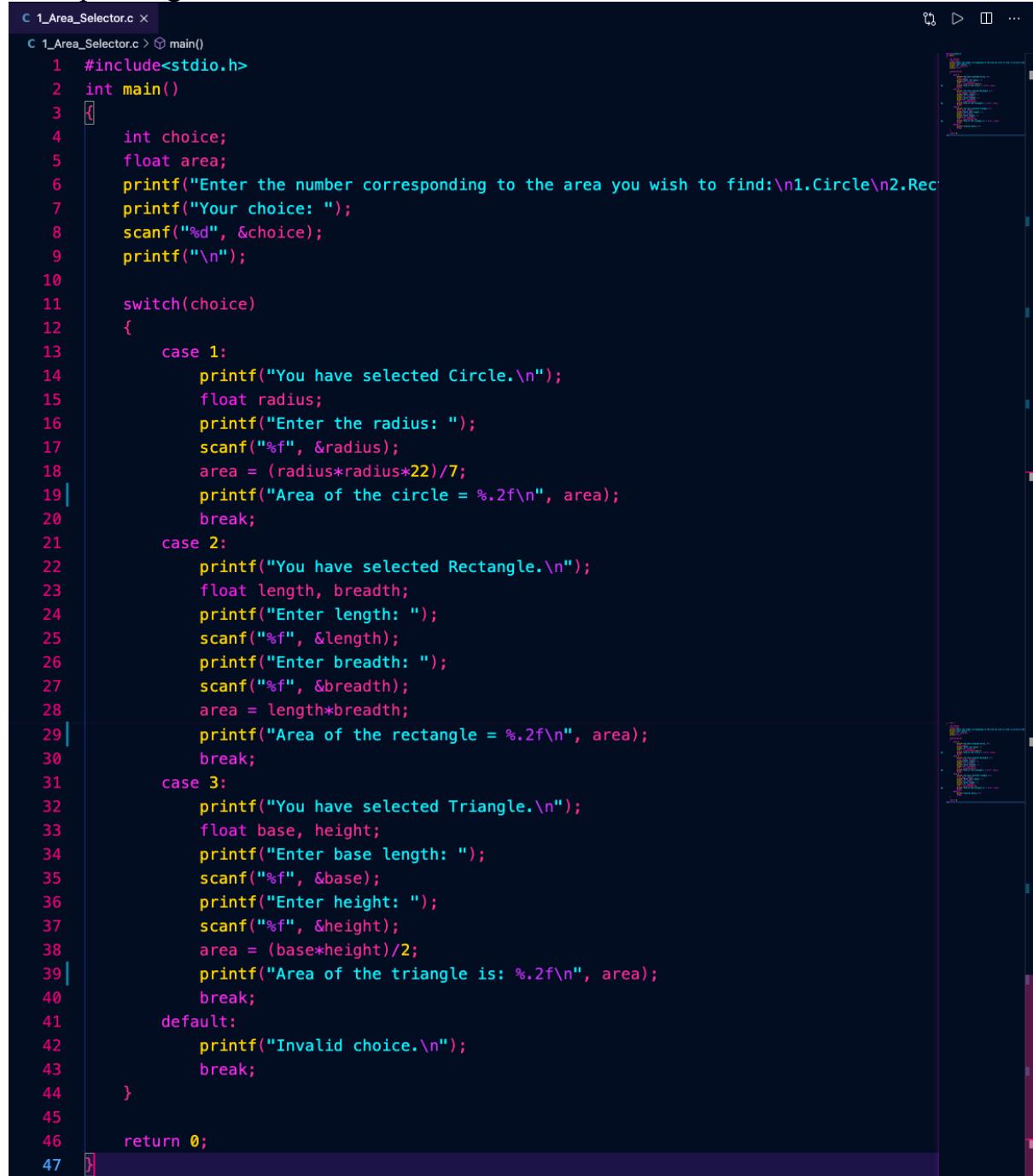
```
C 10_Vowel_Consonant_Digit.c ...
1 #include<stdio.h>
2 int main()
3 {
4     char a;
5     printf("Enter something: ");
6     scanf("%c", &a);
7
8     if(a == 'a' || a == 'e' || a == 'i' || a == 'o' || a == 'u' || a == 'A' || a == 'E' ||
9     {
10         printf("%c is a vowel.\n", a);
11     }
12     else if(a == '1' || a == '2' || a == '3' || a == '4' || a == '5' || a == '6' || a ==
13     {
14         printf("%c is a digit.\n", a);
15     }
16     else if(a == 'b' || a == 'c' || a == 'd' || a == 'f' || a == 'g' || a == 'h' || a ==
17     {
18         printf("%c is a consonant.\n", a);
19     }
20     else
21     {
22         printf("Special character\n");
23     }
24
25     return 0;
26 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ^ x

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: a
a is a vowel.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: E
E is a vowel.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: b
b is a consonant.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: C
C is a consonant.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: 1
1 is a digit.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: 8
8 is a digit.
bumpiestdig10@Rishis-MacBook-Air If_Statement % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" && gcc 10_Vowel_Consonant_Digit.c -o 10_Vowel_Consonant_Digit && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/If_Statement/" 10_Vowel_Consonant_Digit
Enter something: ?
Special character
bumpiestdig10@Rishis-MacBook-Air If_Statement %
```

## Control Structures: Switch-Case

1. Program to calculate an area of a circle, a rectangle or a triangle depending on user's choice.



```

C 1_Area_Selector.c ×
C 1_Area_Selector.c > ⊖ main()
1 #include<stdio.h>
2 int main()
3 {
4     int choice;
5     float area;
6     printf("Enter the number corresponding to the area you wish to find:\n1.Circle\n2.Rectangle\n3.Triangle\n");
7     printf("Your choice: ");
8     scanf("%d", &choice);
9     printf("\n");
10
11    switch(choice)
12    {
13        case 1:
14            printf("You have selected Circle.\n");
15            float radius;
16            printf("Enter the radius: ");
17            scanf("%f", &radius);
18            area = (radius*radius*22)/7;
19            printf("Area of the circle = %.2f\n", area);
20            break;
21        case 2:
22            printf("You have selected Rectangle.\n");
23            float length, breadth;
24            printf("Enter length: ");
25            scanf("%f", &length);
26            printf("Enter breadth: ");
27            scanf("%f", &breadth);
28            area = length*breadth;
29            printf("Area of the rectangle = %.2f\n", area);
30            break;
31        case 3:
32            printf("You have selected Triangle.\n");
33            float base, height;
34            printf("Enter base length: ");
35            scanf("%f", &base);
36            printf("Enter height: ");
37            scanf("%f", &height);
38            area = (base*height)/2;
39            printf("Area of the triangle is: %.2f\n", area);
40            break;
41        default:
42            printf("Invalid choice.\n");
43            break;
44    }
45
46    return 0;
47 }

```

## PROBLEM SOLVING USING COMPUTERS LAB

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + □ □ ^ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 1_Area_Selector
r.c -o 1_Area_Selector && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"1_Area_Selector
bumpiestdig10@Rishi's-MacBook-Air Switch % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 1_Area_Selector.c -o 1_Area_Selector && "/Users/bump
iestdig10/Desktop/College/PSUC/Lab/Switch/"1_Area_Selector
Enter the number corresponding to the area you wish to find:
1.Circle
2.Rectangle
3.Triangle
Your choice: 1

You have selected Circle.
Enter the radius: 5
Area of the circle = 78.57
bumpiestdig10@Rishi's-MacBook-Air Switch % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 1_Area_Selector.c -o 1_Area_Selector && "/Users/bump
iestdig10/Desktop/College/PSUC/Lab/Switch/"1_Area_Selector
Enter the number corresponding to the area you wish to find:
1.Circle
2.Rectangle
3.Triangle
Your choice: 2

You have selected Rectangle.
Enter length: 3.2
Enter breadth: 5.4
Area of the rectangle = 17.28
bumpiestdig10@Rishi's-MacBook-Air Switch % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 1_Area_Selector.c -o 1_Area_Selector && "/Users/bump
iestdig10/Desktop/College/PSUC/Lab/Switch/"1_Area_Selector
Enter the number corresponding to the area you wish to find:
1.Circle
2.Rectangle
3.Triangle
Your choice: 3

You have selected Triangle.
Enter base length: 4.1
Enter height: 2.7
Area of the triangle is: 5.53
bumpiestdig10@Rishi's-MacBook-Air Switch % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 1_Area_Selector.c -o 1_Area_Selector && "/Users/bump
iestdig10/Desktop/College/PSUC/Lab/Switch/"1_Area_Selector
Enter the number corresponding to the area you wish to find:
1.Circle
2.Rectangle
3.Triangle
Your choice: 4

Invalid choice.
bumpiestdig10@Rishi's-MacBook-Air Switch %
```

2. Write a program to design a calculator that performs addition, subtraction, minus and division operation. This program inputs two operands and an operator and then displays the calculated results.

```
C 4_Calculator.c > main()
1 #include <stdio.h>
2 int main()
3 {
4     int choice;
5     float a, b, result;
6     printf("a = ");
7     scanf("%f", &a);
8     printf("b = ");
9     scanf("%f", &b);
10    printf("(1)Addition      (2)Subtraction      (3)Multiplication      (4)Division\n");
11    printf("Enter the operator: ");
12    scanf("%d", &choice);
13    printf("\n");
14
15    switch(choice)
16    {
17        case 1:
18            result = a + b;
19            printf("%.3f + %.3f = %.3f\n", a, b, result);
20            break;
21        case 2:
22            result = a - b;
23            printf("%.3f - %.3f = %.3f\n", a, b, result);
24            break;
25        case 3:
26            result = a * b;
27            printf("%.3f * %.3f = %.3f\n", a, b, result);
28            break;
29        case 4:
30            result = a / b;
31            printf("%.3f/.3f = %.3f\n", a, b, result);
32            break;
33        default: printf("Invalid!\n");
34            break;
35    }
36
37    return 0;
38 }
```

## PROBLEM SOLVING USING COMPUTERS LAB

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + □ ▴ ▲ ×
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 4_Calculator.c -o 4_Calculator && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"4_Calculator
a = 3
b = 9.43
(1)Addition (2)Subtraction (3)Multiplication (4)Division
Enter the operator: 1
3.000 + 9.430 = 12.430
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 4_Calculator.c -o 4_Calculator && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"4_Calculator
a = 45
b = 23.954
(1)Addition (2)Subtraction (3)Multiplication (4)Division
Enter the operator: 2
45.000 - 23.954 = 21.046
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 4_Calculator.c -o 4_Calculator && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"4_Calculator
a = -43
b = 2.766
(1)Addition (2)Subtraction (3)Multiplication (4)Division
Enter the operator: 3
-43.000 * 2.766 = -118.938
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 4_Calculator.c -o 4_Calculator && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"4_Calculator
a = 12.948
b = 7.23
(1)Addition (2)Subtraction (3)Multiplication (4)Division
Enter the operator: 4
12.948/7.230 = 1.791
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/" && gcc 4_Calculator.c -o 4_Calculator && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch/"4_Calculator
a = 23
b = 34.43
(1)Addition (2)Subtraction (3)Multiplication (4)Division
Enter the operator: 5
Invalid!
bumpiestdig10@Rishi's-MacBook-Air ~ %
```

3. Write a program to calculate a bill of internet browsing. The conditions are given below:

Minimum Rs. 200 for up to 100 calls

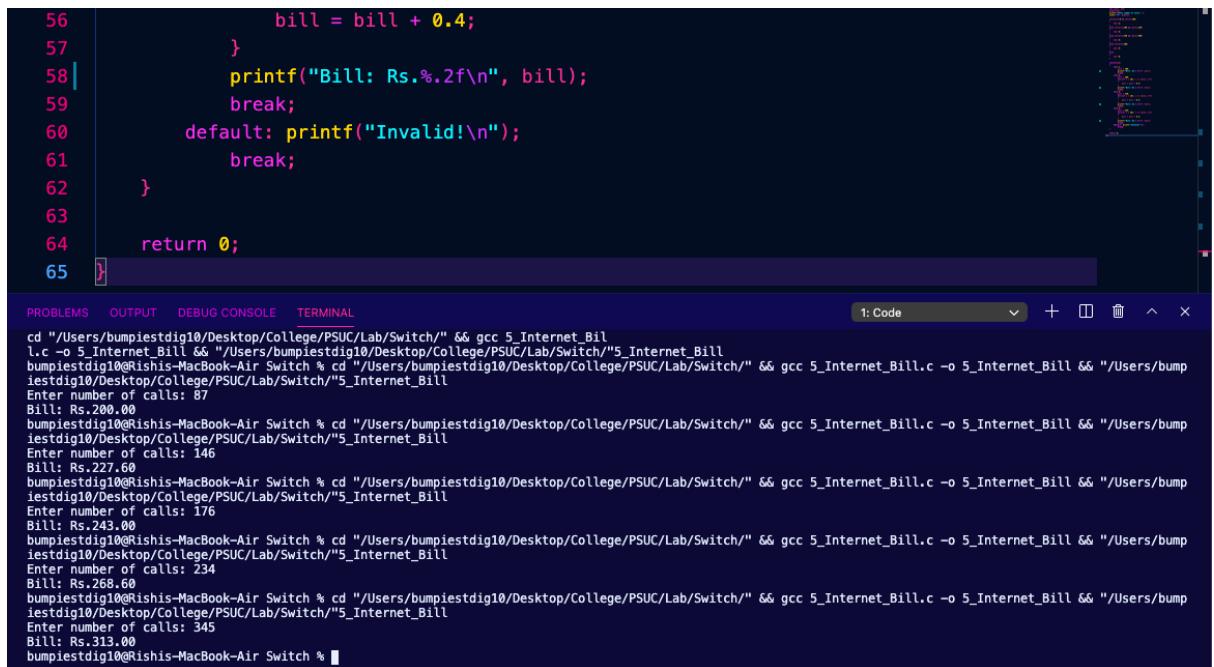
Plus, Rs. 0.60 per call for next 50 calls.

Plus, Rs. 0.50 per call for next 50 calls.

Plus, Rs. 0.40 per call for any call beyond 200 calls

```
c 5_Internet_Bill.c > ⊕ main()
1  #include<stdio.h>
2  int main()
3  {
4      int calls, num;
5      float bill;
6      printf("Enter number of calls: ");
7      scanf("%d", &calls);
8
9      if(calls>=0 && calls<=100)
10     {
11         num =1;
12     }
13     else if(calls>100 && calls<=150)
14     {
15         num =2;
16     }
17     else if(calls>150 && calls<=200)
18     {
19         num =3;
20     }
21     else if(calls>200)
22     {
23         num =4;
24     }
25     else
26     {
27         num =0;
28     }
29
30     switch(num)
31     {
32         case 1:
33             bill = 200;
34             printf("Bill: Rs.%.2f\n", bill);
35             break;
36         case 2:
37             bill = 200;
38             for(int i = 101; i <= calls; i++)
39             {
40                 bill = bill + 0.6;
41             }
42             printf("Bill: Rs.%.2f\n", bill);
43             break;
44         case 3:
45             bill = 230;
46             for(int i = 151; i <= calls; i++)
47             {
48                 bill = bill + 0.5;
49             }
50             printf("Bill: Rs.%.2f\n", bill);
51             break;
52         case 4:
53             bill = 255;
54             for(int i = 201; i <= calls; i++)
55             {
56                 bill = bill + 0.4;
57             }
58     }
59 }
```

## PROBLEM SOLVING USING COMPUTERS LAB



```
56         bill = bill + 0.4;
57     }
58     printf("Bill: Rs.%2f\n", bill);
59     break;
60 default: printf("Invalid!\n");
61     break;
62 }
63
64 return 0;
65 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ 🗑 ^ X

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
Enter number of calls: 87
Bill: Rs.200.00
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
Enter number of calls: 146
Bill: Rs.227.60
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
Enter number of calls: 176
Bill: Rs.243.00
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
Enter number of calls: 234
Bill: Rs.268.60
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch" && gcc 5_Internet_Bill.c -o 5_Internet_Bill && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Switch"/5_Internet_Bill
Enter number of calls: 345
Bill: Rs.313.00
bumpiestdig10@Rishi's-MacBook-Air ~ %
```

## Lab 3. Control Structures: Loops and Nested Loops

1. Write a program to print the sum of N natural numbers.

```

1_Sum_of_N_Natural_Numbers.c > ...
C 1_Sum_of_N_Natural_Numbers.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int n, sum;
5     printf("Enter an integer: ");
6     scanf("%d", &n);
7     sum = 0;
8
9     for(int i = 0; i <= n; i++)
10    {
11        //printf("%d + %d = ", i, sum);
12        sum = sum + i;
13        //printf("%d\n", sum);
14    }
15    printf("Sum of first %d natural numbers: %d\n", n, sum);
16
17    return 0;
18 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" && gcc 1_Sum_of_N_Natural_Numbers.c -o 1_Sum_of_N_Natural_Numbers && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" && gcc 1_Sum_of_N_Natural_Numbers.c -o 1_Sum_of_N_Natural_Numbers && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" 1_Sum_of_N_Natural_Numbers
Enter an integer: 7
Sum of first 7 natural numbers: 28
bumpiestdig10@Rishi's-MacBook-Air Loops %

```

2. Write a program to take N as input print the odd numbers in descending order.

```

2_Odd_Numbers_Descending_Order.c > ...
C 2_Odd_Numbers_Descending_Order.c > ...
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     printf("Enter an integer: ");
6     scanf("%d", &n);
7
8     for(; n >= 0; n--)
9     {
10         if(n%2 != 0)
11         {
12             printf("%d ", n);
13             n = n-1;
14         }
15     }
16     printf("\n");
17
18     return 0;
19 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```

bumpiestdig10@Rishi's-MacBook-Air Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" && gcc 2_Odd_Numbers_Descending_Order.c -o 2_Odd_Numbers_Descending_Order && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" 2_Odd_Numbers_Descending_Order
Enter an integer: 9
9 7 5 3 1
bumpiestdig10@Rishi's-MacBook-Air Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" && gcc 2_Odd_Numbers_Descending_Order.c -o 2_Odd_Numbers_Descending_Order && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Loops/" 2_Odd_Numbers_Descending_Order
Enter an integer: 12
11 9 7 5 3 1
bumpiestdig10@Rishi's-MacBook-Air Loops %

```

## PROBLEM SOLVING USING COMPUTERS LAB

3. Write a program to print the Nth Fibonacci number.

Hint: (Fibonacci series is 0, 1, 1, 2, 3, 5, 8)

The screenshot shows a code editor window with a terminal below it. The code editor displays a C program named `6_Fibonacci.c`. The program prompts the user for a range and then prints the Fibonacci series up to that range. The terminal window shows the execution of the program, including the compilation command `gcc 6_Fibonacci.c -o 6_Fibonacci && ./6_Fibonacci`, the user input `Enter range: 12` and `Enter range: 15`, and the resulting output of the Fibonacci series for each range.

```
 2 int main()
 3 {
 4     int i, range, t1=0, t2=1, nextTerm;
 5     printf("Enter range: ");
 6     scanf("%d", &range);
 7     printf("Fibonacci series: ");
 8
 9     for(i = 1; i <= range; i++)
10    {
11        if(t1 <= range)
12        {
13            printf("%d ", t1);
14            nextTerm = t1 + t2;
15            t1 = t2;
16            t2 = nextTerm;
17        }
18    }
19    printf("\n");
20
21    return 0;
22 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ×

```
bumpiestdig10@Rishi's-MacBook-Air Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Nested_Loops/" && gcc 6_Fibonacci.c -o 6_Fibonacci && "./6_Fibonacci"
Enter range: 12
Fibonacci series: 0 1 1 2 3 5 8
bumpiestdig10@Rishi's-MacBook-Air Nested_Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Nested_Loops/" && gcc 6_Fibonacci.c -o 6_Fibonacci && "./6_Fibonacci"
Enter range: 15
Fibonacci series: 0 1 1 2 3 5 8 13
bumpiestdig10@Rishi's-MacBook-Air Nested_Loops %
```

## 4. Find whether given number is prime or not.



```

C 4_Prime_or_Not.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 3.Loops > C 4_Prime_or_Not.c > main()
1 #include<stdio.h>
2 #include<math.h>
3 int main()
4 {
5     int n;
6     printf("Enter an integer: ");
7     scanf("%d", &n);
8
9     if(n <= 0)
10    {
11        printf("%d is not a prime number.\n", n);
12        return 0;
13    }
14
15    if(n == 1)
16    {
17        printf("%d is neither prime nor composite.\n", n);
18        return 0;
19    }
20
21    for(int i = 2; i <= sqrt(n); i++)
22    {
23        if(n%i == 0)
24        {
25            printf("%d is not a prime number.\n", n);
26            return 0;
27        }
28    }
29
30    printf("%d is a prime number.\n", n);
31
32 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ^ x

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops"
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: 1
1 is neither prime nor composite.
bumpiestdig10@Rishi-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: 7
7 is a prime number.
bumpiestdig10@Rishi-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: 11
11 is not a prime number.
bumpiestdig10@Rishi-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: -21
-21 is not a prime number.
bumpiestdig10@Rishi-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: -3
-3 is not a prime number.
bumpiestdig10@Rishi-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 4_Prime_or_Not.c -o 4_Prime_or_Not && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"4_Prime_or_N
Enter an integer: 0
0 is not a prime number.
bumpiestdig10@Rishi-MacBook-Air 3.Loops %

```

## PROBLEM SOLVING USING COMPUTERS LAB

5. Convert the decimal number into binary to decimal.

$$\text{Ex: } 1101 = 1*2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 13$$

The screenshot shows a code editor window with a dark theme. The file is named `5_Binary_to_Decimal.c`. The code implements a function to convert a binary number to decimal. It uses a loop to iterate through each digit of the binary number, starting from the least significant digit. The decimal value is updated by adding the current digit multiplied by  $2^{i-1}$ , where  $i$  is the position of the digit. The code is annotated with comments explaining the steps. Below the code editor is a terminal window showing the execution of the program and its output.

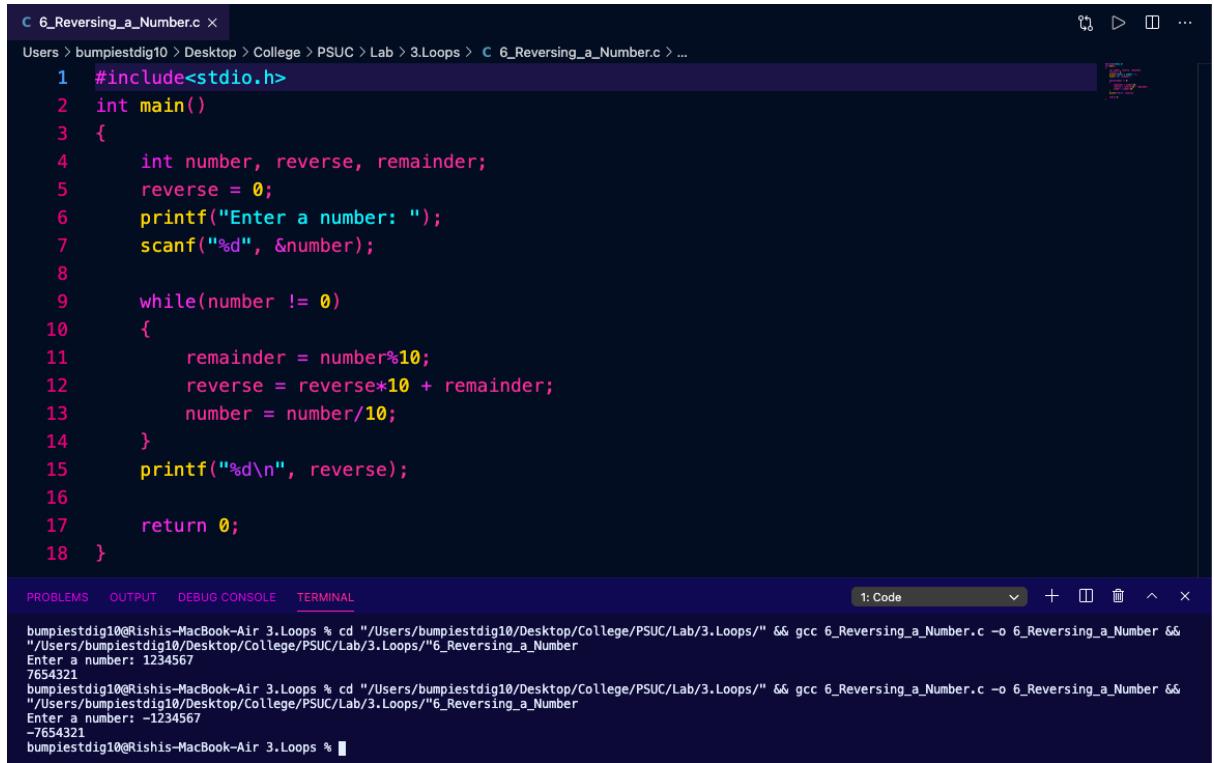
```
c 5_Binary_to_Decimal.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 3.Loops > C 5_Binary_to_Decimal.c > main()
1 #include<stdio.h>
2 #include<math.h>
3
4 int main()
5 {
6     int binary, digits = 0;
7     int decimal;
8
9     /*printf("Number of digits in binary number: ");
10    scanf("%d", &digits);*/
11
12    printf("Enter a binary number: ");
13    scanf("%d", &binary);
14    //printf("\n");
15    decimal = 0;
16
17    int temp = binary;
18
19    while(temp != 0)
20    {
21        digits = digits + 1;
22        temp = temp/10;
23    }
24    //printf("%d\n", digits);
25
26    for(int i = 1; i <= digits; i++)
27    {
28        decimal = decimal + (binary%10)*pow(2, i-1);
29
30        binary = binary/10;
31    }
32    printf("Given binary in decimal: %d\n", decimal);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 5_Binary_to_Decimal.c -o 5_Binary_to_Decimal && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" 5_Binary_to_Decimal
bumpiestdig10@Rishihs-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 5_Binary_to_Decimal.c -o 5_Binary_to_Decimal && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" 5_Binary_to_Decimal
Enter a binary number: 110101
Given binary in decimal: 53
bumpiestdig10@Rishihs-MacBook-Air 3.Loops %
```

## 6. Reverse a given number

Ex: 1234 reverse=4\* $10^3$  + 3 \*  $10^2$  + 2 \*  $10^1$  + 1 \*  $10^0$  = 4321



```

C 6_Reversing_a_Number.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 3.Loops > C 6_Reversing_a_Number.c > ...
1 #include<stdio.h>
2 int main()
3 {
4     int number, reverse, remainder;
5     reverse = 0;
6     printf("Enter a number: ");
7     scanf("%d", &number);
8
9     while(number != 0)
10    {
11         remainder = number%10;
12         reverse = reverse*10 + remainder;
13         number = number/10;
14     }
15     printf("%d\n", reverse);
16
17     return 0;
18 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + □ □ ^ ×
bumpiestdig10@Rishis-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 6_Reversing_a_Number.c -o 6_Reversing_a_Number &&
"/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"6_Reversing_a_Number
Enter a number: 1234567
7654321
bumpiestdig10@Rishis-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 6_Reversing_a_Number.c -o 6_Reversing_a_Number &&
"/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/"6_Reversing_a_Number
Enter a number: -1234567
-7654321
bumpiestdig10@Rishis-MacBook-Air 3.Loops %

```

7. Check whether given number is Armstrong or not. An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. For example, 371 is an Armstrong number since  $3^3 + 7^3 + 1^3 = 371$ .



```

1 #include<stdio.h>
2 #include<math.h>
3 int main()
4 {
5     int number, sum = 0, power, remainder;
6     printf("Enter a number: ");
7     scanf("%d", &number);
8
9     int num, i=0;           //num to check number of places
10    num = number;
11    while(num !=0)
12    {
13        i = i+1;
14        num = num/10;
15    }
16
17    num = number;
18    while(num != 0)
19    {
20        remainder = num%10;
21        power = pow(remainder,i);
22        sum = sum + power;
23        num = num/10;
24    }
25
26    if(sum == number)
27    {
28        printf("%d is an armstrong number.\n", number);
29    }
30    else
31    {
32        printf("%d is not an armstrong number.\n", number);
33    }
34
35    return 0;
36 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ^ X

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 7_Armstrong_Number.c -o 7_Armstrong_Number && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" 7_Armstrong_Number
bumpiestdig10@Rishi's-MacBook-Air ~ cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 7_Armstrong_Number.c -o 7_Armstrong_Number && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" 7_Armstrong_Number
Enter a number: 371
371 is an armstrong number.
bumpiestdig10@Rishi's-MacBook-Air 3.Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" && gcc 7_Armstrong_Number.c -o 7_Armstrong_Number && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Loops/" 7_Armstrong_Number
Enter a number: 369
369 is not an armstrong number.
bumpiestdig10@Rishi's-MacBook-Air 3.Loops %

```

## 8. Print the Fibonacci numbers that fall in given range.

```

C 8_Fibonacci.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 3.Nested_Loops > C 8_Fibonacci.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int i, firstNum, lastNum, t1=0, t2=1, nextTerm;
5     printf("Enter range: ");
6     scanf("%d%d", &firstNum, &lastNum);
7     printf("Fibonacci series in between %d and %d: ", firstNum, lastNum);
8
9     for(i = 1; i <= lastNum; i++)
10    {
11        if(t1 <= lastNum)
12        {
13            if(t1 >= firstNum)
14            {
15                printf("%d ", t1);
16            }
17            nextTerm = t1 + t2;
18            t1 = t2;
19            t2 = nextTerm;
20        }
21    }
22    printf("\n");
23
24    return 0;
25 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ✎ ^ ×

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested_Loops/" && gcc 8_Fibonacci.c -o 8_Fibonacci
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested_Loops/" && gcc 8_Fibonacci.c -o 8_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested_Loops/"8_Fibonacci
Enter range: 0 23
Fibonacci series in between 0 and 23: 0 1 1 2 3 5 8 13 21
bumpiestdig10@Rishis-MacBook-Air 3.Nested_Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested_Loops/" && gcc 8_Fibonacci.c -o 8_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested_Loops/"8_Fibonacci
Enter range: 2 13
Fibonacci series in between 2 and 13: 2 3 5 8 13
bumpiestdig10@Rishis-MacBook-Air 3.Nested_Loops %

```

# PROBLEM SOLVING USING COMPUTERS LAB

9. Print the prime numbers that fall in given range.

C 9\_Prime\_Range.c

Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 3.Nested\_Loops > C 9\_Prime\_Range.c > ...

```
1 #include <stdio.h>
2 int main()
3 {
4     int num1, num2, flag_var, i, j;
5
6     printf("Enter range(input two integer numbers only):\n");
7     scanf("%d %d", &num1, &num2);
8
9     printf("Prime numbers from %d and %d are:\n", num1, num2);
10
11    for(i=num1; i<=num2; i++)
12    {
13        flag_var=0;
14        if(i <= 1)
15        {
16            continue;
17        }
18        for(j=2; j<=i/2; ++j)
19        {
20            if(i%j==0)
21            {
22                flag_var=1;
23                break;
24            }
25        }
26        if(flag_var==0)
27            printf("%d ",i);
28    }
29
30    printf("\n");
31
32    return 0;
33 }
```

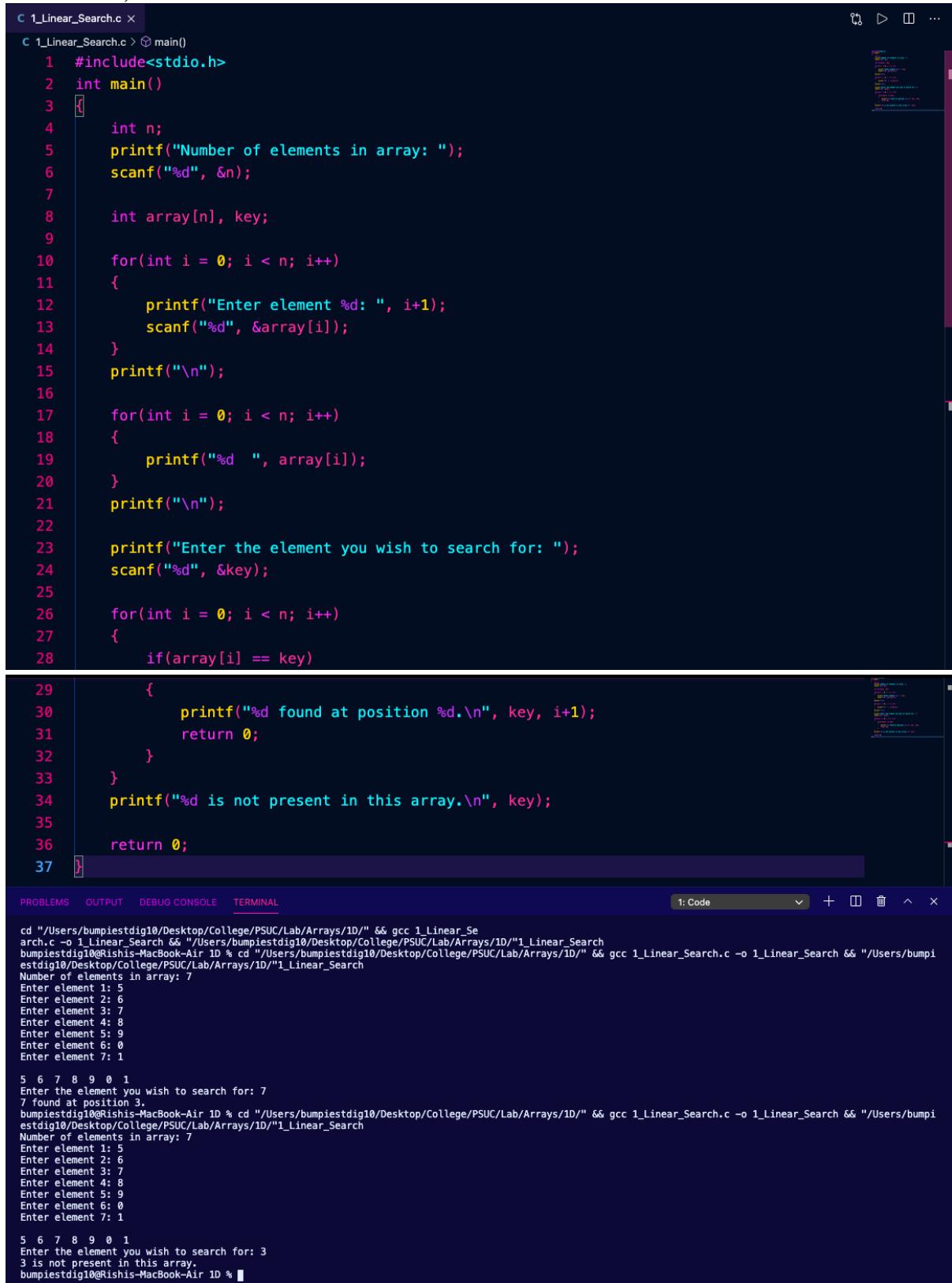
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/" && gcc 9\_Prime\_Range.c -o 9\_Prime\_Range && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/9\_Prime\_Range"
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/" && gcc 9\_Prime\_Range.c -o 9\_Prime\_Range && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/9\_Prime\_Range"
Enter range(input two integer numbers only):
2 81
Prime numbers from 2 and 81 are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79
bumpiestdig10@Rishis-MacBook-Air 3.Nested\_Loops % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/" && gcc 9\_Prime\_Range.c -o 9\_Prime\_Range && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/3.Nested\_Loops/9\_Prime\_Range"
Enter range(input two integer numbers only):
-5 21
Prime numbers from -5 and 21 are:
2 3 5 7 11 13 17 19
bumpiestdig10@Rishis-MacBook-Air 3.Nested\_Loops %

## Lab 4. 1-D and 2-D Array

- Find whether given number is available in an array or not (Linear Search).



```

1_Linear_Search.c X
C 1_Linear_Search.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     printf("Number of elements in array: ");
6     scanf("%d", &n);
7
8     int array[n], key;
9
10    for(int i = 0; i < n; i++)
11    {
12        printf("Enter element %d: ", i+1);
13        scanf("%d", &array[i]);
14    }
15    printf("\n");
16
17    for(int i = 0; i < n; i++)
18    {
19        printf("%d ", array[i]);
20    }
21    printf("\n");
22
23    printf("Enter the element you wish to search for: ");
24    scanf("%d", &key);
25
26    for(int i = 0; i < n; i++)
27    {
28        if(array[i] == key)
29        {
30            printf("%d found at position %d.\n", key, i+1);
31            return 0;
32        }
33    }
34    printf("%d is not present in this array.\n", key);
35
36    return 0;
37

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 1_Linear_Search.c -o 1_Linear_Search && ./1_Linear_Search
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 1_Linear_Search.c -o 1_Linear_Search && ./1_Linear_Search
Number of elements in array: 7
Enter element 1: 5
Enter element 2: 6
Enter element 3: 7
Enter element 4: 8
Enter element 5: 9
Enter element 6: 0
Enter element 7: 1
5 6 7 8 9 0 1
Enter the element you wish to search for: 7
7 found at position 3.
bumpiestdig10@Rishi's-MacBook-Air 1D % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 1_Linear_Search.c -o 1_Linear_Search && ./1_Linear_Search
Number of elements in array: 7
Enter element 1: 5
Enter element 2: 6
Enter element 3: 7
Enter element 4: 8
Enter element 5: 9
Enter element 6: 0
Enter element 7: 1
5 6 7 8 9 0 1
Enter the element you wish to search for: 3
3 is not present in this array.
bumpiestdig10@Rishi's-MacBook-Air 1D %

```

## 2. Find the largest and smallest element in an array

```

C 2_Largest_Smallest.c ×
C 2_Largest_Smallest.c > ...
1 #include<stdio.h>
2 #include<limits.h>
3
4 int main()
5 {
6     int smallest = INT_MAX;
7     //printf("%d", smallest);
8     int largest = INT_MIN;
9
10    int n;
11    printf("Number of elements: ");
12    scanf("%d", &n);
13
14    int array[n];
15    for(int i = 0; i < n; i++)
16    {
17        printf("Element %d: ", i+1);
18        scanf("%d", &array[i]);
19    }
20
21    for(int i = 0; i < n ; i++)
22    {
23        printf("%d ", array[i]);
24    }
25    printf("\n");
26
27    for(int i = 0; i < n; i++)
28    {
29        if(smallest > array[i])
30        {
31            smallest = array[i];
32        }
33
34        if(largest < array[i])
35        {
36            largest = array[i];
37        }
38    }
39
40    printf("Smallest number in the array is: %d\n", smallest);
41    printf("Largest number in the array is: %d\n", largest);
42
43    return 0;
44 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ↻ ⌂ ↻ ×

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 2_Largest_Smallest.c -o 2_Largest_Smallest && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/2_Largest_Smallest"
bumpiestdig10@Rishi's-MacBook-Air 10 % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 2_Largest_Smallest.c -o 2_Largest_Smallest && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/2_Largest_Smallest"
Number of elements: 7
Element 1: 4
Element 2: 6
Element 3: 3
Element 4: 7
Element 5: 34
Element 6: 8
Element 7: 2
4 6 3 7 34 8 2
Smallest number in the array is: 2
Largest number in the array is: 34
bumpiestdig10@Rishi's-MacBook-Air 10 % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 2_Largest_Smallest.c -o 2_Largest_Smallest && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/2_Largest_Smallest"
Number of elements: 7
Element 1: 043
Element 2: -43
Element 3: 54
Element 4: -98
Element 5: 34
Element 6: 56
Element 7: 3
43 -43 54 -98 34 56 3
Smallest number in the array is: -98
Largest number in the array is: 56
bumpiestdig10@Rishi's-MacBook-Air 10 %

```

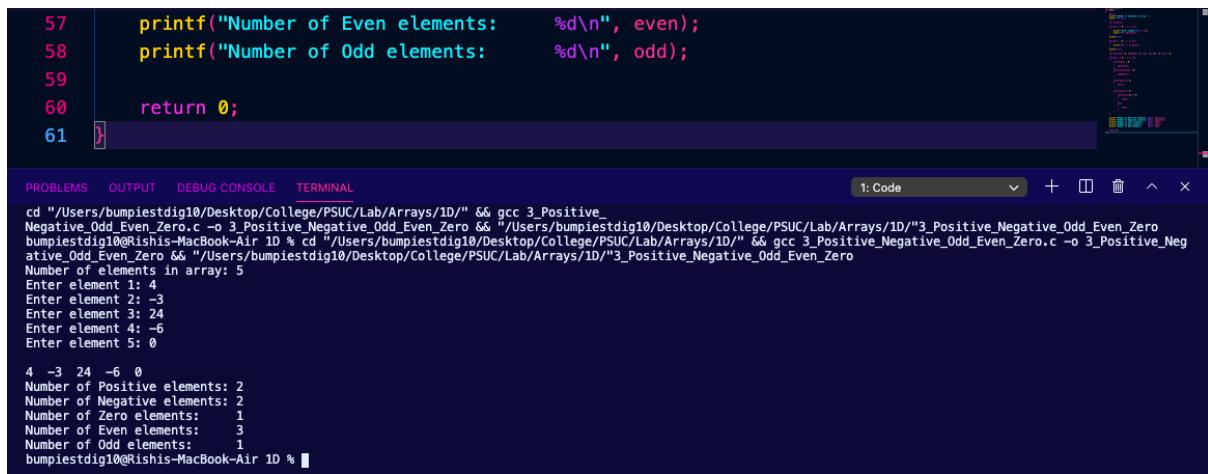
3. Find the number of positive numbers, negative numbers, odd numbers, even numbers and number of 0 of an array.

```

C 3_Positive_Negative_Odd_Even_Zero.c ×
C 3_Positive_Negative_Odd_Even_Zero.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     printf("Number of elements in array: ");
6     scanf("%d", &n);
7
8     int array[n];
9
10    for(int i = 0; i < n; i++)
11    {
12        printf("Enter element %d: ", i+1);
13        scanf("%d", &array[i]);
14    }
15    printf("\n");
16
17    for(int i = 0; i < n; i++)
18    {
19        printf("%d ", array[i]);
20    }
21    printf("\n");
22
23    int positive = 0, negative = 0, even = 0, odd = 0, zero = 0;
24
25    for(int i = 0; i < n; i++)
26    {
27        if(array[i] > 0)
28        {
29            positive++;
30        }
31        else if(array[i] < 0)
32        {
33            negative++;
34        }
35
36        if(array[i] == 0)
37        {
38            zero++;
39        }
40
41        if(array[i] != 0)
42        {
43            if(array[i]%2 == 0)
44            {
45                even++;
46            }
47            else
48            {
49                odd++;
50            }
51        }
52    }
53
54    printf("Number of Positive elements: %d\n", positive);
55    printf("Number of Negative elements: %d\n", negative);
56    printf("Number of Zero elements: %d\n", zero);

```

## PROBLEM SOLVING USING COMPUTERS LAB



The screenshot shows a code editor interface with a dark theme. At the top, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The main area displays the following C code:

```
57     printf("Number of Even elements:      %d\n", even);
58     printf("Number of Odd elements:       %d\n", odd);
59
60     return 0;
61 }
```

Below the code, the terminal window shows the execution of the program. It starts with the command:

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 3_Positive_Negative_Odd_Even_Zero.c -o 3_Positive_Odd_Even_Zero && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" 3_Positive_Negative_Odd_Even_Zero
```

Then it prompts for input:

```
bumpiestdig10@Rishis-MacBook-Air 1D % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 3_Positive_Negative_Odd_Even_Zero.c -o 3_Positive_Negative_Odd_Even_Zero && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" 3_Positive_Negative_Odd_Even_Zero
Number of elements in array: 5
Enter element 1: 4
Enter element 2: -3
Enter element 3: 24
Enter element 4: -6
Enter element 5: 0
```

Finally, it outputs the results:

```
4 -3 24 -6 0
Number of Positive elements: 2
Number of Negative elements: 2
Number of Zero elements: 1
Number of Even elements: 3
Number of Odd elements: 1
bumpiestdig10@Rishis-MacBook-Air 1D %
```

4. Reverse an array with an auxiliary array.

The screenshot shows a code editor with a dark theme. The main area displays a C program named `4_Reverse_an_Array.c`. The code prompts the user for the number of elements in an array, then asks for each element, stores them in a temporary array, and finally prints the reversed array. The code is color-coded for readability.

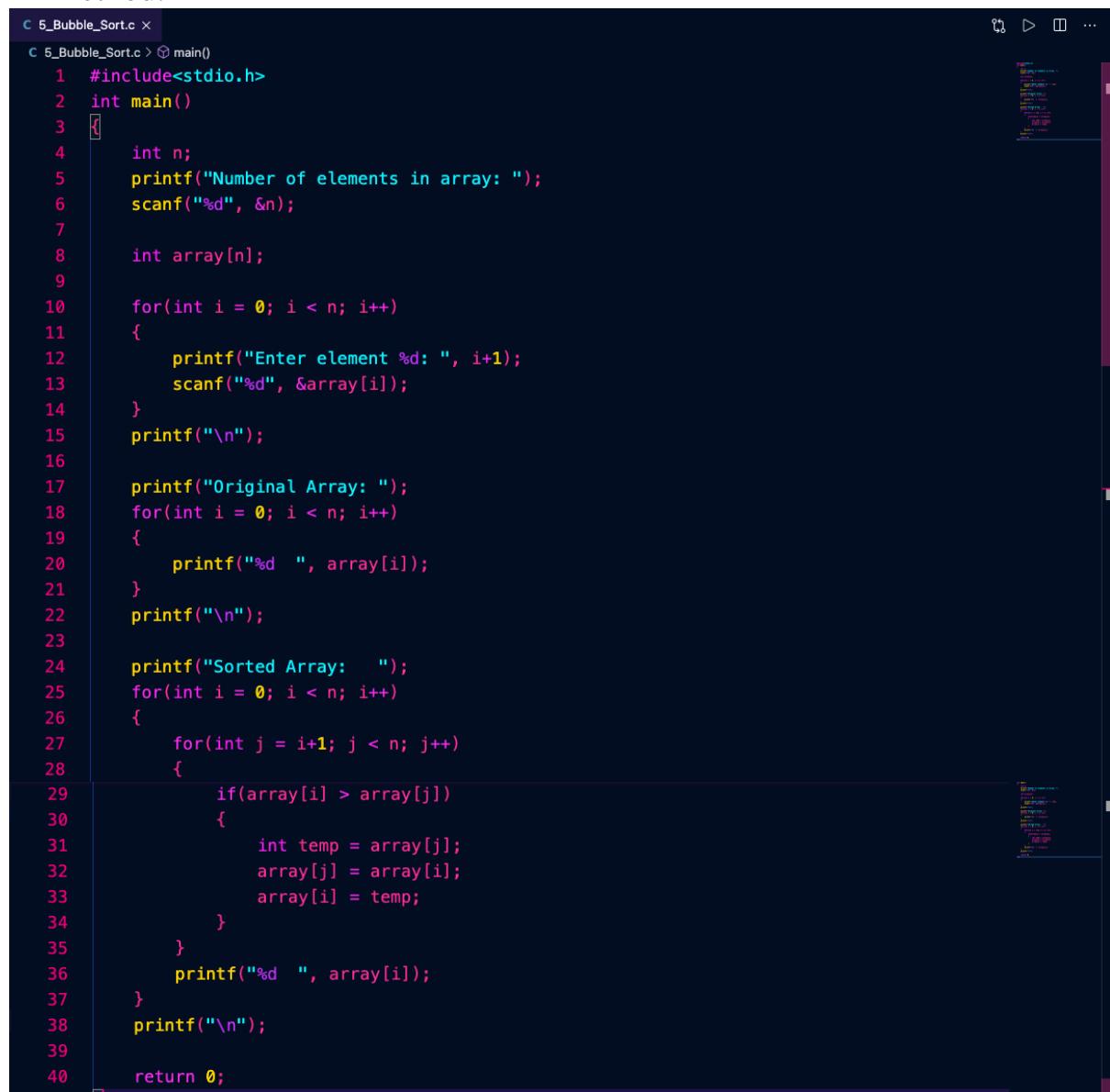
```
C 4_Reverse_an_Array.c ×
C 4_Reverse_an_Array.c > ⚡ main()
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     printf("Number of elements in array: ");
6     scanf("%d", &n);
7
8     int array[n];
9
10    for(int i = 0; i < n; i++)
11    {
12        printf("Enter element %d: ", i+1);
13        scanf("%d", &array[i]);
14    }
15    printf("\n");
16
17    printf("Original Array: ");
18    for(int i = 0; i < n; i++)
19    {
20        printf("%d ", array[i]);
21    }
22    printf("\n");
23
24    int reverse[n];
25
26    printf("Reversed Array: ");
27    for(int i = n-1; i >= 0; i--)
28    {
29        reverse[i] = array[n-i-1];
30    }
31
32    for(int i = 0; i < n; i++)
33    {
34        printf("%d ", reverse[i]);
35    }
36    printf("\n");
37
38    return 0;
39 }
```

Below the code editor, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is active, showing the command-line output of the program's execution. The output shows the user entering 8 elements (1 through 8) and the program printing the original array and its reversed version.

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 4_Reverse_an_Array.c -o 4_Reverse_an_Array && ./4_Reverse_an_Array
bumpiestdig10@Rishi's-MacBook-Air 1D % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 4_Reverse_an_Array.c -o 4_Reverse_an_Array && ./4_Reverse_an_Array
Number of elements in array: 8
Enter element 1: 3
Enter element 2: -6
Enter element 3: 0
Enter element 4: 7
Enter element 5: 2
Enter element 6: 0
Enter element 7: -2
Enter element 8: 5

Original Array: 3 -6 0 7 2 0 -2 5
Reversed Array: 5 -2 0 2 7 0 -6 3
bumpiestdig10@Rishi's-MacBook-Air 1D %
```

5. Arrange the elements of an array in ascending order by bubble sort method.



```

C 5_Bubble_Sort.c ×
C 5_Bubble_Sort.c > ⌂ main()
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     printf("Number of elements in array: ");
6     scanf("%d", &n);
7
8     int array[n];
9
10    for(int i = 0; i < n; i++)
11    {
12        printf("Enter element %d: ", i+1);
13        scanf("%d", &array[i]);
14    }
15    printf("\n");
16
17    printf("Original Array: ");
18    for(int i = 0; i < n; i++)
19    {
20        printf("%d ", array[i]);
21    }
22    printf("\n");
23
24    printf("Sorted Array: ");
25    for(int i = 0; i < n; i++)
26    {
27        for(int j = i+1; j < n; j++)
28        {
29            if(array[i] > array[j])
30            {
31                int temp = array[j];
32                array[j] = array[i];
33                array[i] = temp;
34            }
35        }
36        printf("%d ", array[i]);
37    }
38    printf("\n");
39
40    return 0;
41 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ⌂ ^ ×

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 5_Bubble_Sort.c -o 5_Bubble_Sort && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/5_Bubble_Sort"
bumpiestdig10@Rishis-MacBook-Air 1D % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/" && gcc 5_Bubble_Sort.c -o 5_Bubble_Sort && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/Arrays/1D/5_Bubble_Sort"
Number of elements in array: 5
Enter element 1: 2
Enter element 2: 6
Enter element 3: 0
Enter element 4: -3
Enter element 5: -8
Original Array: 2 6 0 -3 -8
Sorted Array: -8 -3 0 2 6
bumpiestdig10@Rishis-MacBook-Air 1D %

```

## Using 2-D Array

6. Write a program to take 3 student marks of 5 subjects. Print the total marks of each student and average marks of each subject.

```

6_Marks.c ×
2D > C 6_Marks.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     int marks[3][5];
6
7     for(int i = 0; i < 3; i++)
8     {
9         for(int j = 0; j < 5; j++)
10        {
11            printf("Enter marks of subject %d for student %d: ", j+1, i+1);
12            scanf("%d", &marks[i][j]);
13        }
14        printf("\n");
15    }
16
17    int total;
18    float average;
19
20    for(int i = 0; i < 3; i++)
21    {
22        total = 0;
23        for(int j = 0; j < 5; j++)
24        {
25            total = total + marks[i][j];
26            average = total/5.0;
27        }
28        printf("Total marks of student %d: %d\n", i+1, total);
29        printf("Average marks of student %d: %.1f\n\n", i+1, average);
30    }
31
32    return 0;
33 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + □ □ ^ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" && gcc 6_Marks.c
-o 6_Marks && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" 6_Marks
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" && gcc 6_Marks.c -o 6_Marks && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" 6_Marks
Enter marks of subject 1 for student 1: 100
Enter marks of subject 2 for student 1: 89
Enter marks of subject 3 for student 1: 68
Enter marks of subject 4 for student 1: 70
Enter marks of subject 5 for student 1: 97

Enter marks of subject 1 for student 2: 58
Enter marks of subject 2 for student 2: 67
Enter marks of subject 3 for student 2: 54
Enter marks of subject 4 for student 2: 65
Enter marks of subject 5 for student 2: 66

Enter marks of subject 1 for student 3: 46
Enter marks of subject 2 for student 3: 57
Enter marks of subject 3 for student 3: 98
Enter marks of subject 4 for student 3: 96
Enter marks of subject 5 for student 3: 97

Total marks of student 1: 424
Average marks of student 1: 84.8

Total marks of student 2: 310
Average marks of student 2: 62.0

Total marks of student 3: 394
Average marks of student 3: 78.8
bumpiestdig10@Rishi's-MacBook-Air 2D %

```

## 7. Multiplication of two matrices.

```

C 7_Matrix_Multiplication.c ✘
C 7_Matrix_Multiplication.c > ⌂ main()
1 #include<stdio.h>
2
3 int main()
4 {
5     int rA, rB, cA, cB;
6
7     //Taking number of rows and columns as input
8     printf("Enter the number of rows in A : ");
9     scanf("%d", &rA);
10    printf("Enter the number of columns in A: ");
11    scanf("%d", &cA);
12    printf("Enter the number of rows in B : ");
13    scanf("%d", &rB);
14    printf("Enter the number of columns in B: ");
15    scanf("%d", &cB);
16    printf("\n");
17
18    //Checking if Multiplication can be done or not
19    if(cA != rB)
20    {
21        printf("Number of columns of A must be equal to number of rows of B.\n");
22        return 0;
23    }
24
25    //Declaring A and B
26    float A[rA][cA], B[rB][cB];
27
28    //Initializing A
29    for(int i = 0; i < rA; i++)
30    {
31        for(int j = 0; j < cA; j++)
32        {
33            printf("Enter A%d%d: ", i+1, j+1);
34            scanf("%f", &A[i][j]);
35        }
36    }
37    printf("\n");
38
39    //Initializing B
40    for(int i = 0; i < rB; i++)
41    {
42        for(int j = 0; j < cB; j++)
43        {
44            printf("Enter B%d%d: ", i+1, j+1);
45            scanf("%f", &B[i][j]);
46        }
47    }
48    printf("\n");
49
50    //Printing A
51    printf("A:\n");
52    for(int i = 0; i < rA; i++)
53    {
54        for(int j = 0; j < cA; j++)
55        {
56            printf("%.2f    ", A[i][j]);
57        }
58    }
59
60    //Printing B
61    printf("B:\n");
62    for(int i = 0; i < rB; i++)
63    {
64        for(int j = 0; j < cB; j++)
65        {
66            printf("%.2f    ", B[i][j]);
67        }
68    }
69
70    //Multiplication
71    for(int i = 0; i < rA; i++)
72    {
73        for(int j = 0; j < cB; j++)
74        {
75            float sum = 0;
76            for(int k = 0; k < rB; k++)
77            {
78                sum += A[i][k] * B[k][j];
79            }
80            printf("%.2f    ", sum);
81        }
82    }
83
84    //Printing Result
85    printf("Result:\n");
86    for(int i = 0; i < rA; i++)
87    {
88        for(int j = 0; j < cB; j++)
89        {
90            printf("%.2f    ", result[i][j]);
91        }
92    }
93
94    //End
95    return 0;
96}

```

## PROBLEM SOLVING USING COMPUTERS LAB

```
57     }
58     printf("\n");
59 }
printf("\n");

//Printing B
printf("B:\n");
for(int i = 0; i < rB; i++)
{
    for(int j = 0; j < cB; j++)
    {
        printf("%.2f    ", B[i][j]);
    }
    printf("\n");
}
printf("\n");

//Declaring C
float C[rA][cB];

//Initializing C (Multiplying A and B)
for(int i = 0; i < rA; i++)
{
    for(int j = 0; j < cB; j++)
    {
        C[i][j] = 0;
        for(int k = 0; k < cA; k++)
        {
            C[i][j] = C[i][j] + (A[i][k] * B[k][j]);
        }
    }
}

//Printing C
printf("A * B = C:\n");
for(int i = 0; i < rA; i++)
{
    for(int j = 0; j < cB; j++)
    {
        printf("%.2f    ", C[i][j]);
    }
    printf("\n");
}
printf("\n");

102     return 0;
103 }
```

## PROBLEM SOLVING USING COMPUTERS LAB

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + ⌂ ⌂ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/" && gcc 7_Matrix_Multiplication.c -o 7_Matrix_Multiplication && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/" 7_Matrix_Multiplication
bumpiestdig10@Rishis-MacBook-Air 4.Arrays % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/" && gcc 7_Matrix_Multiplication.c -o 7_Matrix_Multiplication
Enter the number of rows in A : 2
Enter the number of columns in A: 3
Enter the number of rows in B : 3
Enter the number of columns in B: 2

Enter A11: -2
Enter A12: 4.3
Enter A13: 0
Enter A21: 1
Enter A22: 3.22
Enter A23: 3

Enter B11: 7
Enter B12: -3.5
Enter B13: 2
Enter B21: 0
Enter B31: 5
Enter B32: 1

A:
-2.00    4.30    0.00
1.00    3.22    3.00

B:
7.00    -3.50
2.00    0.00
5.00    1.00

A * B = C:
-5.40    7.00
28.44   -0.50

bumpiestdig10@Rishis-MacBook-Air 4.Arrays % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/" && gcc 7_Matrix_Multiplication.c -o 7_Matrix_Multiplication && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/" 7_Matrix_Multiplication
Enter the number of rows in A : 2
Enter the number of columns in A: 3
Enter the number of rows in B : 2
Enter the number of columns in B: 3

Number of columns of A must be equal to number of rows of B.
bumpiestdig10@Rishis-MacBook-Air 4.Arrays %
```

8. Find whether a given matrix is symmetric or not. (Hint:  $A = A^T$ )

```

2D > C 8_Symmetric_Matrix.c > main()
1  #include<stdio.h>
2
3  int main()
4  {
5      int rA, cA;
6
7      //Taking number of rows and columns as input.
8      printf("Number of rows      :");
9      scanf("%d", &rA);
10     printf("Number of columns   :");
11     scanf("%d", &cA);
12     printf("\n");
13
14     //Checking if the matrix is square matrix or not
15     if(rA != cA)
16     {
17         printf("It cannot be a symmetric matrix.\n");
18         return 0;
19     }
20
21     //Declaring Matrix
22     float A[rA][cA];
23
24     //Initializing Matrix
25     for(int i = 0; i < rA; i++)
26     {
27         for(int j = 0; j < cA; j++)
28         {
29             printf("A%d%d: ", i+1, j+1);
30             scanf("%f", &A[i][j]);
31         }
32     }
33     printf("\n");
34
35     //Printing Matrix
36     for(int i = 0; i < rA; i++)
37     {
38         for(int j = 0; j < cA; j++)
39         {
40             printf("%.2f    ", A[i][j]);
41         }
42         printf("\n");
43     }
44     printf("\n");
45
46     //Checking Symmetry
47     for(int i = 0; i < rA; i++)
48     {
49         for(int j = 0; j < cA; j++)
50         {
51             if(A[i][j] != A[j][i])
52             {
53                 printf("Matrix is not symmetric.\n");
54                 return 0;
55             }
56         }
57     }
58     printf("Matrix is symmetric.\n");
59
60     return 0;
61 }
```

# PROBLEM SOLVING USING COMPUTERS LAB

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" && gcc 8_Symmetric_Matrix.c -o 8_Symmetric_Matrix && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" 8_Symmetric_Matrix
bumpiestdig10@Rishi's-MacBook-Air 4.Arrays % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" && gcc 8_Symmetric_Matrix.c -o 8_Symmetric_Matrix && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" 8_Symmetric_Matrix
Number of rows :2
Number of columns :3

It cannot be a symmetric matrix.
bumpiestdig10@Rishi's-MacBook-Air 2D % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" && gcc 8_Symmetric_Matrix.c -o 8_Symmetric_Matrix && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/4.Arrays/2D/" 8_Symmetric_Matrix
Number of rows :3
Number of columns :3

A11: 1
A12: 2
A13: 3
A21: 2
A22: 4
A23: 5
A31: 3
A32: 5
A33: 6

1.00  2.00  3.00
2.00  4.00  5.00
3.00  5.00  6.00

Matrix is symmetric.
bumpiestdig10@Rishi's-MacBook-Air 2D %
```

## Lab 5. Strings

1. Print number of vowels and consonant in a string.

```
1_Vowels_Consonants.c X
C 1_Vowels_Consonants.c > ⚡ main()
1 #include<stdio.h>
2
3 int main()
4 {
5     char str[100];
6     int vowels = 0, consonants = 0;
7
8     printf("Enter a string: ");
9     fgets(str, sizeof(str), stdin);
10
11    for(int i = 0; i < 100; i++)
12    {
13        if(str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] ==
14        {
15            vowels++;
16        }
17        else if(str[i] == 'b' || str[i] == 'c' || str[i] == 'd' || str[i] == 'f' || str[i]
18        {
19            consonants++;
20        }
21
22    }
23
24    printf("The string has %d vowel(s) and %d consonant(s).\n", vowels, consonants);
25
26    return 0;
27 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
cd "/Users/bumpiestdig10/Desktop/Collage/PSUC/Lab/5.Strings/" && gcc 1_Vowels_Consonants.c -o 1_Vowels_Consonants && "Users/bumpiestdig10/Desktop/Collage/PSUC/Lab/5.Strings/"1_Vowels_Consonants
bumpiestdig10@Rishi-MacBook-Air:5.Strings % cd "/Users/bumpiestdig10/Desktop/Collage/PSUC/Lab/5.Strings/" && gcc 1_Vowels_Consonants.c -o 1_Vowels_Consonants &
& "/Users/bumpiestdig10/Desktop/Collage/PSUC/Lab/5.Strings/"1_Vowels_Consonants
Enter a string: Rishi Bothra PSUC LAB
The string has 6 vowel(s) and 12 consonants.
bumpiestdig10@Rishi-MacBook-Air:5.Strings %
```

## PROBLEM SOLVING USING COMPUTERS LAB

### 2. Change all lower-case letters into an upper-case in a sentence.

The screenshot shows a code editor window with a dark theme. The file being edited is `2_Upper_Case.c`. The code uses `stdio.h` and `ctype.h` headers. It prompts the user to enter a string, reads it into `str`, and then iterates through each character, applying `toupper` to convert it to uppercase before printing the result. The terminal below shows the execution of the program and its output.

```
C 2_Upper_Case.c > main()
1 #include<stdio.h>
2 #include<ctype.h>
3
4 int main()
5 {
6     char str[100];
7
8     printf("Enter a string: ");
9     fgets(str, sizeof(str), stdin);
10
11    for(int i = 0; i < 100; i++)
12    {
13        str[i] = toupper(str[i]);
14    }
15    puts(str);
16
17
18 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + X

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 2_Upper_Case.c -o 2_Upper_Case && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/">2_Upper_Case
bumpiestdig10@Rishi-MacBook-Air 5.Strings % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 2_Upper_Case.c -o 2_Upper_Case && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/">2_Upper_Case
Enter a string: Rishi Bothra
RISHI BOTHRA
bumpiestdig10@Rishi-MacBook-Air 5.Strings %
```

### 3. Count the number of words in a sentence.

The screenshot shows a code editor window with a dark theme. The file being edited is `3_Words.c`. The code uses `stdio.h`. It prompts the user to enter a string, reads it into `str`, and then counts the number of words by incrementing a counter whenever it encounters a space, newline, or tab character. The terminal below shows the execution of the program and its output.

```
C 3_Words.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     char str[100];
6
7     printf("Enter a string: ");
8     fgets(str, sizeof(str), stdin);
9
10    int counter = 0, i = 0;
11
12    while(str[i] != '\0')
13    {
14        if(str[i] == ' ' || str[i] == '\n' || str[i] == '\t')
15        {
16            counter++;
17        }
18        i++;
19    }
20
21    printf("There are %d word(s) in this string.\n", counter);
22
23
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + X

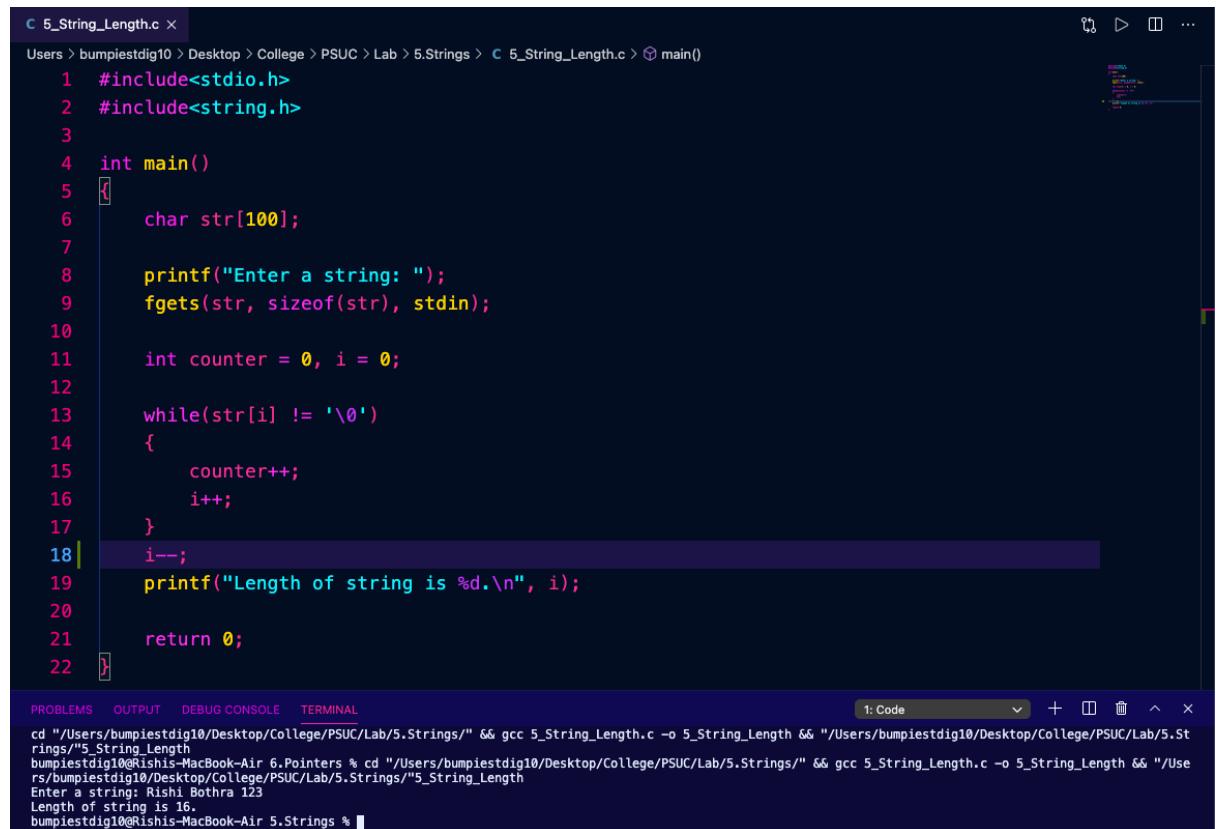
```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 3_Words.c -o 3_Words && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/">3_Words
bumpiestdig10@Rishi-MacBook-Air 5.Strings % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 3_Words.c -o 3_Words && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/">3_Words
Enter a string: Rishi Bothra PSUC LAB
There are 4 word(s) in this string.
bumpiestdig10@Rishi-MacBook-Air 5.Strings %
```

4. Reverse a string.

The screenshot shows a terminal window with the following content:

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 4_Reverse.c -o 4_Reverse && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" 4_Reverse
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 4_Reverse.c -o 4_Reverse && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" "4_Reverse"
Enter a string:
Rishi Bothra PSUC LAB
Reverse of the string:
BAL CUSP arhtoB ihsir
bumpiestdig10@Rishi-MacBook-Air ~ %
```

5. Find the string length of a string without using the predefined strlen() function.



```

C 5_String_Length.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 5.Strings > C 5_String_Length.c > main()
1 #include<stdio.h>
2 #include<string.h>
3
4 int main()
5 {
6     char str[100];
7
8     printf("Enter a string: ");
9     fgets(str, sizeof(str), stdin);
10
11    int counter = 0, i = 0;
12
13    while(str[i] != '\0')
14    {
15        counter++;
16        i++;
17    }
18    i--;
19    printf("Length of string is %d.\n", i);
20
21    return 0;
22 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ^ X

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 5_String_Length.c -o 5_String_Length && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/"<"/5_String_Length"
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 5_String_Length.c -o 5_String_Length && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/"<"/5_String_Length"
Enter a String: Rishi Bothra 123
Length of string is 16.
bumpiestdig10@Rishi-MacBook-Air 5.Strings %

```

## 6. Find the sub-string of a given string.

```

C 6_Substring.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 5.Strings > C 6_Substring.c > main()
1 #include <stdio.h>
2
3 int main()
4 {
5     char str1[80], str2[80];
6     int l, i, j;
7
8     printf("Enter first string: ");
9     gets(str1);
10
11    printf("Enter second string: ");
12    gets(str2);
13
14 //finding length of second string
15    for (l = 0; str2[l] != '\0'; l++);
16
17    for (i = 0, j = 0; str1[i] != '\0' && str2[j] != '\0'; i++)
18    {
19        if (str1[i] == str2[j])
20        {
21            j++;
22        }
23        else
24        {
25            j = 0;
26        }
27    }
28
29    if (j == l)
30    {
31        printf("Substring found at position %d\n", i - j + 1);
32    }
33    else
34    {
35        printf("Substring not found\n");
36    }
37
38
39

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 6_Substring
.c -o 6_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/"6_Substring
bumpiestdig10@Rishi-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 6_Substring.c -o 6_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/"6_Substring
warning: this program uses gets(), which is unsafe.
Enter first string: Rishi Bothra is great
Enter second string: Bothra
Substring found at position 7
bumpiestdig10@Rishi-MacBook-Air 5.Strings % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 6_Substring.c -o 6_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/"6_Substring
warning: this program uses gets(), which is unsafe.
Enter first string: Rishi Bothra is great
Enter second string: Naman
Substring not found
bumpiestdig10@Rishi-MacBook-Air 5.Strings %

```

7. Check whether the given string is a palindrome or not.

```
C 7_Palindrome.c x
C 7_Palindrome.c > main()
1 #include<stdio.h>
2 #include<string.h>
3
4 int main()
5 {
6     char s[100];
7
8     printf("Enter the string: ");
9     gets(s);
10
11    int n = strlen(s);
12    int i, c = 0;
13
14    for(i = 0; i < n/2; i++)
15    {
16        if(s[i] == s[n-i-1])
17        {
18            c++;
19        }
20    }
21
22    if(c == i)
23    {
24        printf("\'%s\' is a palindrome.\n", s);
25    }
26    else
27    {
28        printf("\'%s\' is not a palindrome.\n", s);
29    }
30
31    return 0;
32 }
```

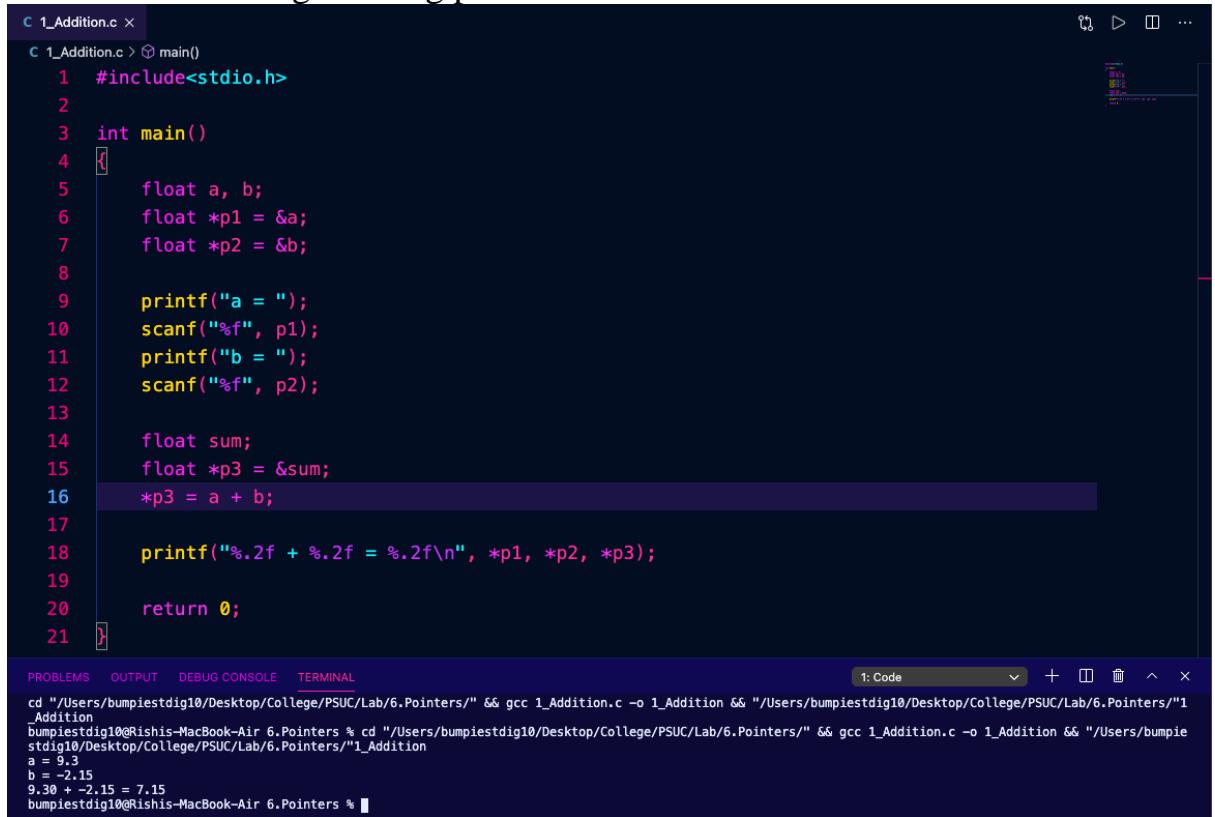
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 7_Palindrome.c -o 7_Palindrome
bumpiestdig10@Rishi's-MacBook-Air 5.Strings % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 7_Palindrome.c -o 7_Palindrome && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" 7_Palindrome
warning: this program uses gets(), which is unsafe.
Enter the string: Rishi Bothra
"Rishi Bothra" is not a palindrome.
bumpiestdig10@Rishi's-MacBook-Air 5.Strings % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" && gcc 7_Palindrome.c -o 7_Palindrome && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/5.Strings/" 7_Palindrome
warning: this program uses gets(), which is unsafe.
Enter the string: MOM
"MON" is a palindrome.
bumpiestdig10@Rishi's-MacBook-Air 5.Strings %
```

## Lab 6. Pointers

- Access two integers using pointers and add them.



```

1 1_Addition.c ×
C 1_Addition.c > ⊖ main()
1 #include<stdio.h>
2
3 int main()
4 {
5     float a, b;
6     float *p1 = &a;
7     float *p2 = &b;
8
9     printf("a = ");
10    scanf("%f", p1);
11    printf("b = ");
12    scanf("%f", p2);
13
14    float sum;
15    float *p3 = &sum;
16    *p3 = a + b;
17
18    printf("%.2f + %.2f = %.2f\n", *p1, *p2, *p3);
19
20    return 0;
21 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ^ X

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 1_Addition.c -o 1_Addition && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/"1_Addition
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 1_Addition.c -o 1_Addition && "/Users/bumpie
stdig10/Desktop/College/PSUC/Lab/6.Pointers/"1_Addition
a = 9.3
b = -2.15
9.30 + -2.15 = 7.15
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers %

```

2. Write a program to find out the greatest and the smallest among the three numbers using pointers.

```

C 2_Greatest.c ×
C 2_Greatest.c > ⌂ main()
1 #include<stdio.h>
2
3 int main()
4 {
5     int a, b, c;
6     int *p1 = &a, *p2 = &b, *p3 = &c;
7
8     printf("a = ");
9     scanf("%d", &a);
10    printf("b = ");
11    scanf("%d", &b);
12    printf("c = ");
13    scanf("%d", &c);
14
15    if(*p1 > *p2 && *p1 > *p3)
16    {
17        printf("%d is greatest.\n", *p1);
18    }
19    else if(*p2 > *p1 && *p2 > *p3)
20    {
21        printf("%d is greatest.\n", *p2);
22    }
23    else
24    {
25        printf("%d is greatest\n", *p3);
26    }
27
28    return 0;
29

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + 🖍️ ^ X

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 2_Greatest
.c -o 2_Greatest && ./2_Greatest
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 2_Greatest.c -o 2_Greatest && "/Users/bumpie
stdig10/Desktop/College/PSUC/Lab/6.Pointers/"2_Greatest
a = 5
b = 6
c = 7
7 is greatest
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 2_Greatest.c -o 2_Greatest && "/Users/bumpie
stdig10/Desktop/College/PSUC/Lab/6.Pointers/"2_Greatest
a = -2
b = -7
c = -4
-2 is greatest.
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 2_Greatest.c -o 2_Greatest && "/Users/bumpie
stdig10/Desktop/College/PSUC/Lab/6.Pointers/"2_Greatest
a = 0
b = 4
c = -2
4 is greatest.
bumpiestdig10@Rishi-MacBook-Air ~ %

```

### 3. Determine the length of a character string using a pointer.

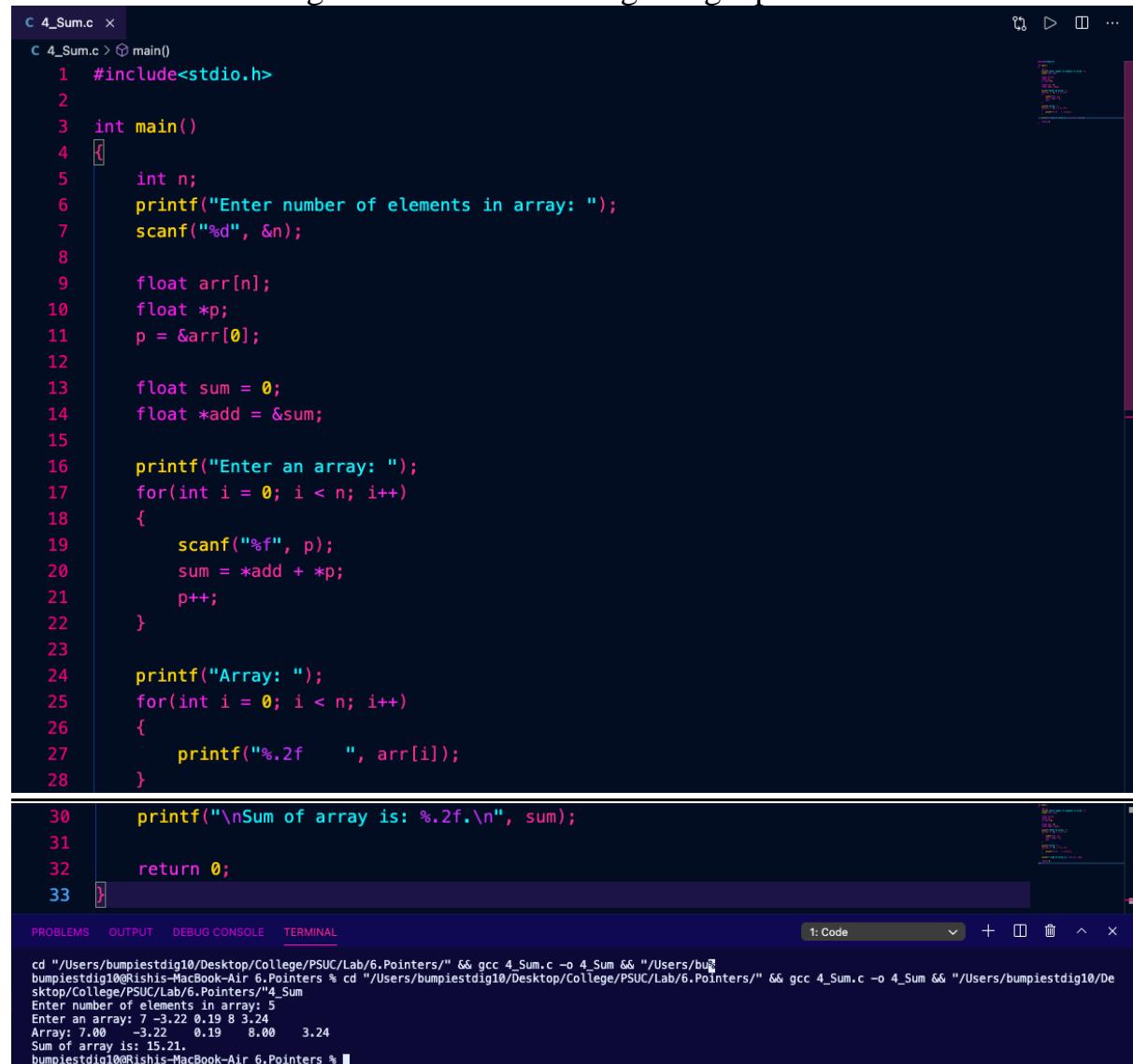


```
C 3_Length.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     char s[100];
6     char *p;
7     p = &s[0];
8
9     printf("Enter a string: ");
10    fgets(s, sizeof(s), stdin);
11
12    int length = 0;
13    while(*p != '\0')
14    {
15        length++;
16        p++;
17    }
18
19    length--;
20
21    printf("Length of string is: %d.\n", length);
22
23    return 0;
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + ☰ ^ ×

```
bumpiestdig10@Rishi-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 3_Length.c -o 3_Length && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/"3_Length
Enter a string: Rishi Bothra PSUC LAB 6
Length of string is: 23.
bumpiestdig10@Rishi-MacBook-Air 6.Pointers %
```

## 4. Determine the length of a character string using a pointer.



```

C 4_Sum.c ×
C 4_Sum.c > main()
1 #include<stdio.h>
2
3 int main()
4 {
5     int n;
6     printf("Enter number of elements in array: ");
7     scanf("%d", &n);
8
9     float arr[n];
10    float *p;
11    p = &arr[0];
12
13    float sum = 0;
14    float *add = &sum;
15
16    printf("Enter an array: ");
17    for(int i = 0; i < n; i++)
18    {
19        scanf("%f", p);
20        sum = *add + *p;
21        p++;
22    }
23
24    printf("Array: ");
25    for(int i = 0; i < n; i++)
26    {
27        printf("%.2f    ", arr[i]);
28    }
29
30    printf("\nSum of array is: %.2f.\n", sum);
31
32    return 0;
33

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code

```

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 4_Sum.c -o 4_Sum && "/Users/bumpiestdig10@Rishi's-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 4_Sum.c -o 4_Sum && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers"/"4_Sum
Enter number of elements in array: 5
Enter an array: 7 -3.22 0.19 8 3.24
Array: 7.00 -3.22 0.19 8.00 3.24
Sum of array is: 15.21.
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers %

```

5. Determine whether a substring (string 1) is in the main sting or not. If present, return the pointer of the first occurrence.

```

C 5_Substring.c > ⊖ main()
1  #include<stdio.h>
2
3  int main()
4  {
5      char str[80], search[80];
6
7      char *ptr1;
8      ptr1 = &str[0];
9      char *ptr2;
10     ptr2 = &search[0];
11
12     int count1 = 0, count2 = 0, i, j, flag;
13
14     printf("Enter a string: ");
15     gets(str);
16     printf("Enter search substring: ");
17     gets(search);
18
19     while (*ptr1 != '\0')
20     {
21         count1++;
22         ptr1++;
23     }
24
25     while (*ptr2 != '\0')
26     {
27         count2++;
28         ptr2++;
29     }
30
31     for (i = 0; i <= count1 - count2; i++)
32     {
33         for (j = i; j < i + count2; j++)
34         {
35             flag = 1;
36             if (str[j] != search[j - i])
37             {
38                 flag = 0;
39                 break;
40             }
41         }
42         if (flag == 1)
43             break;
44     }
45
46     if (flag == 1)
47         printf("SEARCH SUCCESSFUL!\nFirst occurrence at position: %d.\n", i+1);
48     else
49         printf("SEARCH UNSUCCESSFUL!\n");
50
51     return 0;
52 }
```

## PROBLEM SOLVING USING COMPUTERS LAB

The screenshot shows a terminal window with the following text:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: Code + □ ▴ ^ ×
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 5_Substring.c -o 5_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/">5_Substring
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 5_Substring.c -o 5_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/">5_Substring
warning: this program uses gets(), which is unsafe.
Enter a string: Rishi Bothra is great
Enter search substring: Bot
SEARCH SUCCESSFUL!
First occurrence at position: 7.
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/" && gcc 5_Substring.c -o 5_Substring && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/6.Pointers/">5_Substring
warning: this program uses gets(), which is unsafe.
Enter a string: Rishi Bothra is great
Enter search substring: RIB
SEARCH UNSUCCESSFUL!
bumpiestdig10@Rishi's-MacBook-Air 6.Pointers %
```

## Lab 7. Functions

1. Find the factorial of a number using a function.

```
c 1_Factorial.c <
C 1_Factorial.c > factorial(int)
1 #include<stdio.h>
2
3 int factorial(int n);
4 int product = 1;
5
6 int main()
7 {
8     int num;
9     printf("Enter a positive integer: ");
10    scanf("%d", &num);
11
12    if(num < 0)
13    {
14        printf("Invalid Input!\n");
15        return 0;
16    }
17
18    printf("%d! = ", num);
19    factorial(num);
20    printf("%d\n", product);
21
22    return 0;
23 }
24
25 int factorial(int n)
26 [
27     if(n == 0)
28     {
29         return product;
30     }
31
32     for(int i = 1; i <= n; i++)
33     {
34         product = product * i;
35     }
36
37     return product;
38 ]
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

```
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 1_Factorial.c -o 1_Factorial && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"1_Factorial
Enter a positive integer: 7
7! = 5040
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 1_Factorial.c -o 1_Factorial && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"1_Factorial
Enter a positive integer: 0
0! = 1
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 1_Factorial.c -o 1_Factorial && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"1_Factorial
Enter a positive integer: -3
Invalid Input!
bumpiestdig10@Rishis-MacBook-Air ~ %
```

## PROBLEM SOLVING USING COMPUTERS LAB

2. Find the maximum of a given set of numbers using functions.  
Find GCD of two numbers recursively.

The screenshot shows a code editor window with the following details:

- Title Bar:** C 2\_Greatest\_GCD.c ×
- File Path:** Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 7.Functions > C 2\_Greatest\_GCD.c > hcf(int, int)
- Code Content:**

```
1 #include <stdio.h>
2
3 int hcf(int n1, int n2);
4
5 int main()
6 {
7     int n1, n2;
8
9     printf("n1 = ");
10    scanf("%d", &n1);
11    printf("n2 = ");
12    scanf("%d", &n2);
13
14    printf("\nG.C.D of %d and %d is %d.\n", n1, n2, hcf(n1, n2));
15    return 0;
16 }
17
18 int hcf(int n1, int n2)
19 {
20     if (n2 != 0)
21     {
22         return hcf(n2, n1 % n2);
23     }
24     else
25     {
26         return n1;
27     }
28 }
```
- Terminal Output:**

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 2_Greatest_GCD.c -o 2_Greatest_GCD
bumpiestdig10@Rishi's-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 2_Greatest_GCD.c -o 2_Greatest_GCD && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"2_Greatest_GCD
n1 = 68
n2 = 33

G.C.D of 68 and 33 is 1.

bumpiestdig10@Rishi's-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 2_Greatest_GCD.c -o 2_Greatest_GCD && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"2_Greatest_GCD
n1 = 21
n2 = 84

G.C.D of 21 and 84 is 21.

bumpiestdig10@Rishi's-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 2_Greatest_GCD.c -o 2_Greatest_GCD && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"2_Greatest_GCD
n1 = 28
n2 = 60

G.C.D of 28 and 60 is 4.

bumpiestdig10@Rishi's-MacBook-Air 7.Functions %
```

3. Check whether the given number is prime or not. Using this function generate first n prime numbers using the above function.

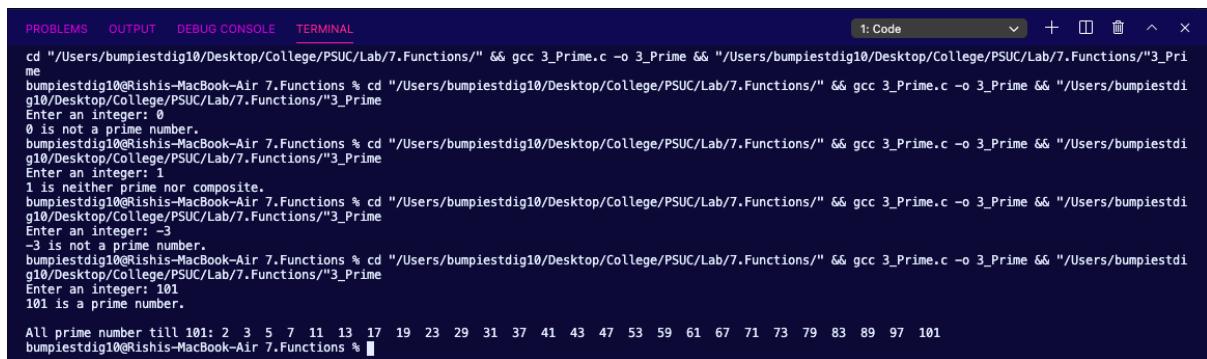


```

C 3_Prime.c x
C 3_Prime.c > ⊖ main()
1 #include<stdio.h>
2 #include<math.h>
3
4 int Prime(int x);
5
6 int main()
7 {
8     int n;
9     printf("Enter an integer: ");
10    scanf("%d", &n);
11
12    if(n <= 0)
13    {
14        printf("%d is not a prime number.\n", n);
15        return 0;
16    }
17    else if(n == 1)
18    {
19        printf("%d is neither prime nor composite.\n", n);
20        return 0;
21    }
22    else if(Prime(n) == 0)
23    {
24        printf("%d is not a prime number.\n\n", n);
25    }
26    else
27    {
28        printf("%d is a prime number.\n\n", n);
29    }
30
31    printf("All prime number till %d: ", n);
32    for(int i = 2; i <= n; i++)
33    {
34        if(Prime(i) != 0)
35        {
36            printf("%d ", i);
37        }
38    }
39    printf("\n");
40
41    return 0;
42 }
43
44 int Prime(int x)
45 {
46     for(int i = 2; i <= sqrt(x); i++)
47     {
48         if(x % i == 0)
49         {
50             return 0;
51         }
52     }
53     return x;
54 }

```

## PROBLEM SOLVING USING COMPUTERS LAB



The screenshot shows a terminal window with the following session:

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 3_Prime.c -o 3_Prime && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/">"3_Prime"
me
bumpiestdig10@Rishis-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 3_Prime.c -o 3_Prime && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/">"3_Prime
Enter an integer: 0
0 is not a prime number.
bumpiestdig10@Rishis-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 3_Prime.c -o 3_Prime && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/">"3_Prime
Enter an integer: 1
1 is neither prime nor composite.
bumpiestdig10@Rishis-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 3_Prime.c -o 3_Prime && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/">"3_Prime
Enter an integer: -3
-3 is not a prime number.
bumpiestdig10@Rishis-MacBook-Air 7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 3_Prime.c -o 3_Prime && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/">"3_Prime
Enter an integer: 101
101 is a prime number.

All prime number till 101: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101
bumpiestdig10@Rishis-MacBook-Air 7.Functions %
```

# PROBLEM SOLVING USING COMPUTERS LAB

4. Write a function to generate the nth Fibonacci term using recursion. Print first N Fibonacci terms using this function.

C 4\_Fibonacci.c ×

```
1 #include <stdio.h>
2 int fibo(int);
3
4 int main()
5 {
6     int num;
7
8     printf("Enter the nth number in fibonacci series: ");
9     scanf("%d", &num);
10
11    if (num < 0)
12    {
13        printf("Invalid input!\n");
14    }
15    else
16    {
17        printf("The %d number in fibonacci series is %d\n\n", num, fibo(num));
18
19        printf("Fibonacci series for %d terms:\n", num);
20        for(int i = 1; i <= num; i++)
21        {
22            printf("%d ", fibo(i));
23        }
24        printf("\n");
25    }
26
27    return 0;
28 }
29
30 int fibo(int num)
31 {
32     if (num == 0)
33     {
34         return 0;
35     }
36     else if (num == 1)
37     {
38         return 1;
39     }
40     else
41     {
42         return(fibo(num - 1) + fibo(num - 2));
43     }
44 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

```
cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 4_Fibonacci.c -o 4_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"4_Fibonacci
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 4_Fibonacci.c -o 4_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"4_Fibonacci
Enter the nth number in fibonacci series: -4
Invalid input!
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 4_Fibonacci.c -o 4_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"4_Fibonacci
Enter the nth number in fibonacci series: 0
The 0 number in fibonacci series is 0

Fibonacci series for 0 terms:
```

```
bumpiestdig10@Rishis-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 4_Fibonacci.c -o 4_Fibonacci && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"4_Fibonacci
Enter the nth number in fibonacci series: 8
The 8 number in fibonacci series is 21

Fibonacci series for 8 terms:
1 1 2 3 5 8 13 21
bumpiestdig10@Rishis-MacBook-Air ~ %
```

5. Check if the given string is a palindrome or not, using string handling function.

C 5\_Palindrome.c ×

```
5_Palindrome.c > ⌂ Palindrome(int, char [n])
 1 #include<stdio.h>
 2 #include<string.h>
 3
 4 void Palindrome(int n, char str[n]);
 5
 6 int main()
 7 {
 8     /*int n;
 9     printf("Length of the string: ");
10     scanf("%d", &n);*/
11     char str[1000];
12     printf("Enter a string: ");
13     gets(str);
14     int n = strlen(str);
15
16     char *ptr;
17     ptr = &str[0];
18
19     Palindrome(n, ptr);
20
21     return 0;
22 }
23
24 void Palindrome(int n, char str[n])
25 {
26     int c = 0, i;
27     n = strlen(str);
28
29     for(i = 0; i < n/2; i++)
30     {
31         if(str[i] == str[n-i-1])
32         {
33             c++;
34         }
35     }
36
37     if (c == i)
38     {
39         printf("String is a palindrome.\n");
40     }
41     else
42     {
43         printf("String is not a palindrome.\n");
44     }
45 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

1: Code

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 5\_Palindr  
ome.c -o 5\_Palindrome && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"5\_Palindrome  
bumpiestdig10@Rishihs-MacBook-Air:7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 5\_Palindrome.c -o 5\_Palindrome && "/Users/  
bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"5\_Palindrome  
warning: this program uses gets(), which is unsafe.  
Enter a string: rishi  
String is not a palindrome.  
bumpiestdig10@Rishihs-MacBook-Air:7.Functions % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 5\_Palindrome.c -o 5\_Palindrome && "/Users/  
bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/"5\_Palindrome  
warning: this program uses gets(), which is unsafe.  
Enter a string: mom  
String is a palindrome.  
bumpiestdig10@Rishihs-MacBook-Air:7.Functions %

6. Write a function **Sort** for sorting a list of names which will use a function **compare** to compare two names. (bubble Sort may be used).



```

C 6_Name.c ×
Users > bumpiestdig10 > Desktop > College > PSUC > Lab > 7.Functions > C 6_Name.c > ...
1 #include<stdio.h>
2 #include<string.h>
3
4 char name[25][25];
5 int count;
6 int i, j;
7
8 void sort(int count);
9 int compare();
10
11 int main()
12 {
13     printf("Number of names: ");
14     scanf("%d", &count);
15
16     printf("Enter names one by one:\n");
17     for(i = 0; i <= count; i++)
18     {
19         fgets(name[i], sizeof(name[i]), stdin);
20     }
21     sort(count);
22
23     printf("\nSorted name: ");
24     for(i = 0; i <= count; i++)
25     {
26         fputs(name[i], stdout);
27     }
28
29     return 0;
30 }
31
32 void sort(int count)
33 {
34     char temp[25];
35
36     for(i = 0; i <= count; i++)
37     {
38         for(j = i+1; j <= count; j++)
39         {
40             if(compare() > 0)
41             {
42                 strcpy(temp, name[i]);
43                 strcpy(name[i], name[j]);
44                 strcpy(name[j], temp);
45             }
46         }
47     }
48 }
49
50 int compare()
51 {
52     return strcmp(name[i], name[j]);
53 }

```

PROBLEM SOLVING USING COMPUTERS LAB

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Code + □ ×

cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 6_Name.c
-o 6_Name && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" 6_Name
bumpiestdig10@Rishi-MacBook-Air ~ % cd "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" && gcc 6_Name.c -o 6_Name && "/Users/bumpiestdig10/Desktop/College/PSUC/Lab/7.Functions/" 6_Name
Number of names: 5
Enter names one by one:
Rishi
Charitra
Simran
Mir
Riya

Sorted name:
Charitra
Mir
Rishi
Riya
Simran
bumpiestdig10@Rishi-MacBook-Air 7.Functions %
```