GRAPHIC ERA

DEEMED TO BE UNIVERSITY

INTRODUCTION TO C PROGRAMMING

BATCH: (2023-2026)

B.C.A. 1ST YEAR

SUBMITTED BY. **SUBMITTED TO.**

Vinay Kumar

MR. RISHI KUMAR

ST.ID. 2316012158

ASSISTANT PROFESSOR:-

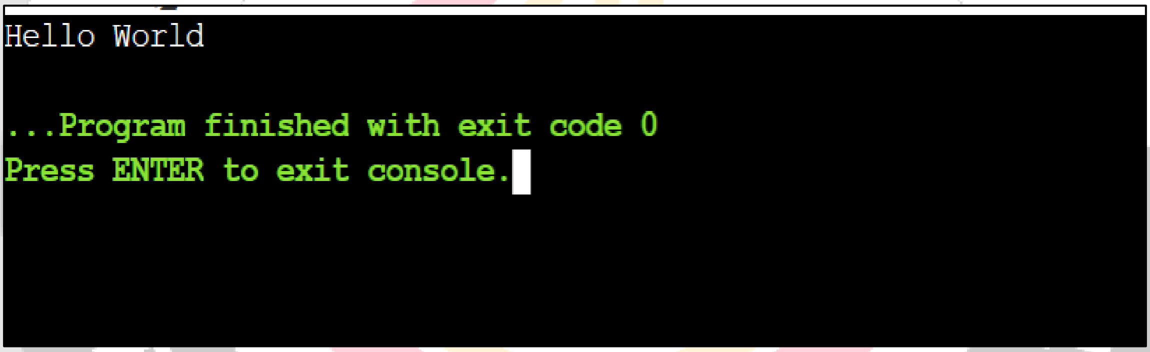
POOJA CHAHAR

CSIT , GEU

1. Write a program to print "Hello World".

```
#include <stdio.h>
int main() {
printf ("Hello World");
      return 0;
}
//Vinay Kumar
```

OUTPUT:

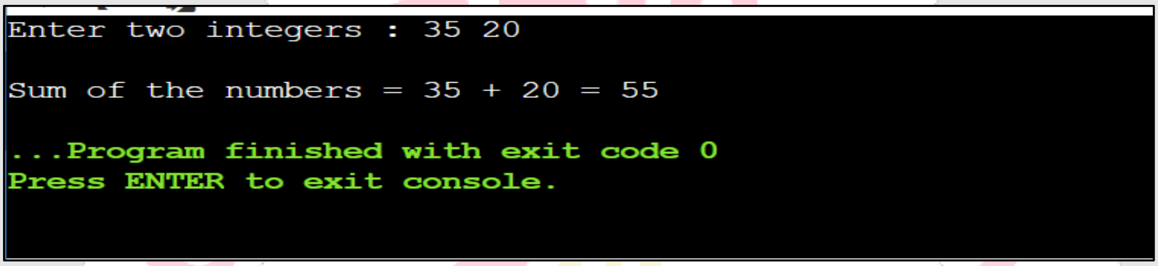


```
Hello World

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to add two numbers

```
#include <stdio.h>
int main() {
int num1, num2, sum;
printf ("Enter two integers");
scanf ("%d %d", &num1, &num2);
      //calculate the sum
      sum = num1 + num2;
printf ("\nSum of the numbers =%d + %d = %d", num1, num2, sum);
      return 0;
}
//Vinay Kumar
```

OUTPUT:

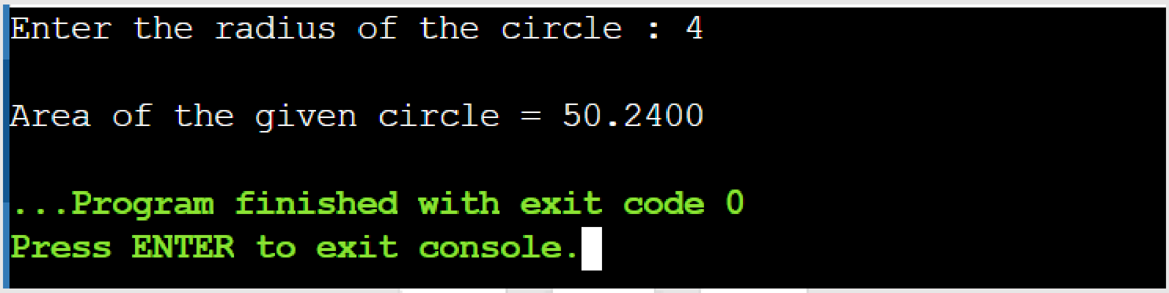


```
Enter two integers : 35 20

Sum of the numbers = 35 + 20 = 55

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to find area of circle.

```
#include <stdio.h>
#define pie 3.14
int main() {
    float radius, area;
printf ("Enter the radius of the circle : ");
scanf ("%f", &radius);
    area = pie * radius * radius;
printf ("Area of the given circle = %.4f", area);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

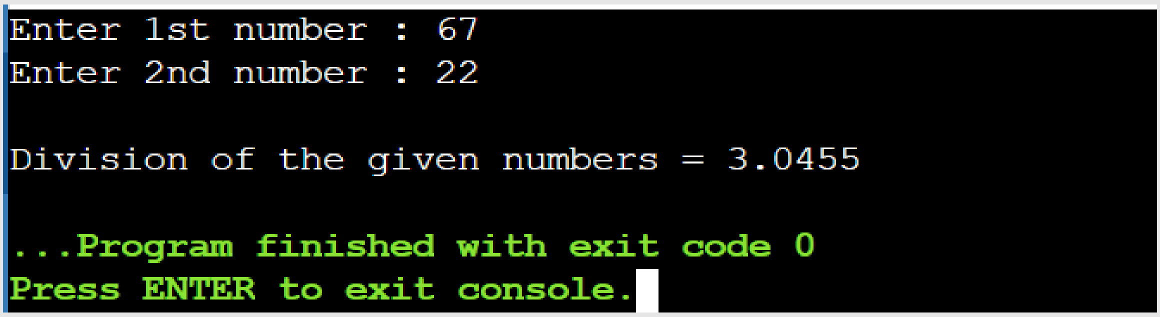


```
Enter the radius of the circle : 4

Area of the given circle = 50.2400

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to divide two numbers.

```
#include <stdio.h>
int main() {
    float a, b;
    float div;
printf ("Enter 1st number : ");
scanf ("%f", &a);
printf ("Enter 2nd number : ");
scanf ("%f", &b);
    div = a / b;
printf ("\nDivision of the given numbers = %.4f", div);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

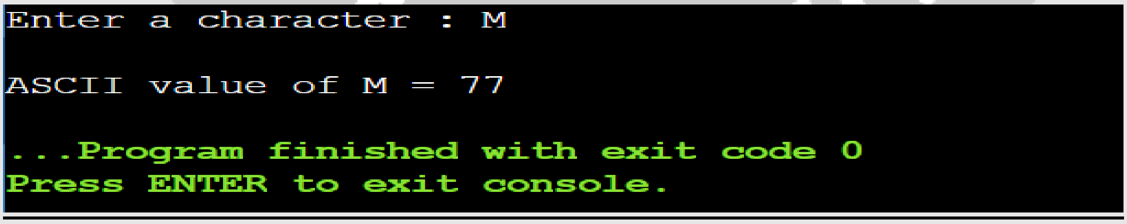


```
Enter 1st number : 67
Enter 2nd number : 22

Division of the given numbers = 3.0455

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to print ASCII value.

```
#include <stdio.h>
int main() {
    char c;
printf ("Enter a character : ");
scanf ("%c", &c);
    //%d display the integer value of a character
    //%c display the actual character
printf ("\nASCII value of %c = %d", c, c);
    return 0;
}
//Vinay Kumar
```

OUTPUT:



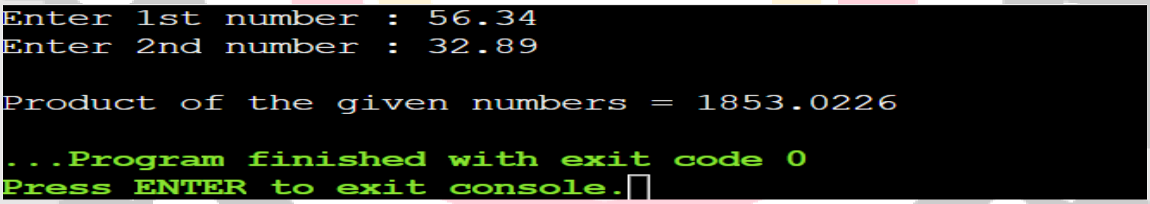
```
Enter a character : M

ASCII value of M = 77

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to multiply floating numbers.

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

```
int main() {
    double a, b, product;
printf ("Enter 1st number : ");
scanf ("%lf", &a);
printf ("Enter 2nd number : ");
scanf ("%lf", &b);
    product = a * b;
printf ("\nProduct of the given numbers = %.4f", product);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

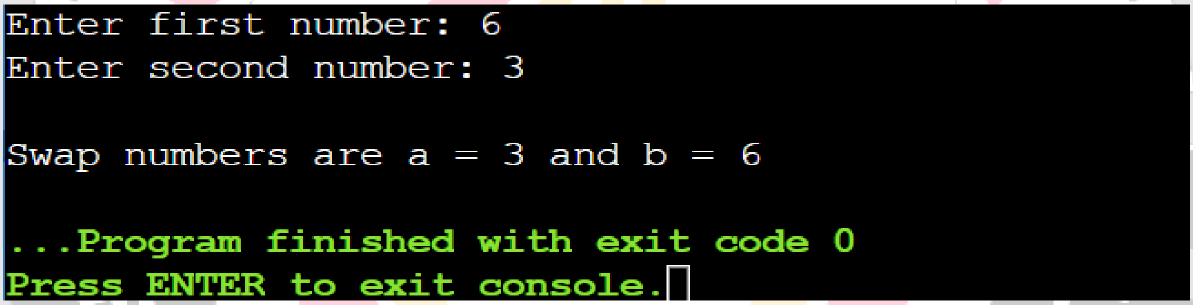


```
Enter 1st number : 56.34
Enter 2nd number : 32.89

Product of the given numbers = 1853.0226

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to SWAP two variables by using third variable.

```
#include <stdio.h>
Int main() {
int a, b, c;
printf("Enter first number: ");
scanf ("%d", &a);
printf ("Enter second number: ");
scanf ("%d", &b);
    c = a;
    a = b;
    b = c;
printf ("\nSwap numbers are a = %d and b = %d", a, b);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

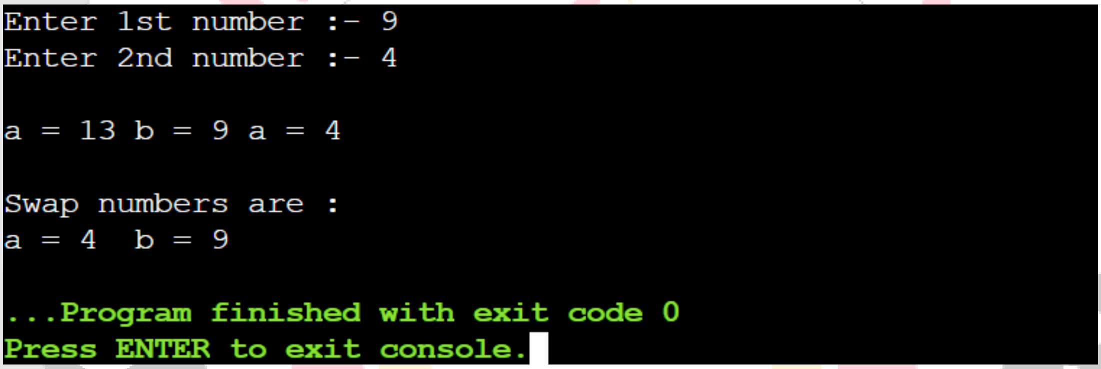


```
Enter first number: 6
Enter second number: 3

Swap numbers are a = 3 and b = 6

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to SWAP two variables without using third variable.

```
#include <stdio.h>
int main () {
        int a, b;
        printf ("Enter 1st number :- ");
        scanf ("%d", &a);
        printf ("Enter 2nd number :- ");
        scanf ("%d", &b);
        a = a + b; /*a = %d+%d = %d*/ printf ("\na = %d", a);
        b = a - b;  /*b = %d-%d = %d*/ printf (" b = %d", b);
        a = a - b;  /*a = %d-%d = %d*/ printf (" a = %d\n", a);
        printf ("\nSwap numbers are :\na = %d  \nb = %d", a, b);
        return 0;
}
//Vinay Kumar
```

OUTPUT:

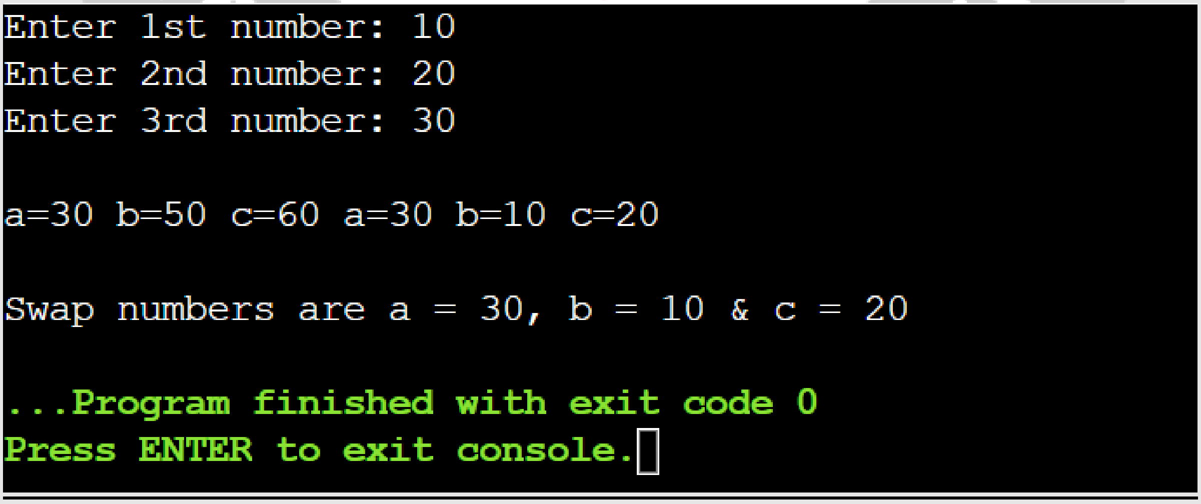


```
Enter 1st number :- 9
Enter 2nd number :- 4

a = 13 b = 9 a = 4

Swap numbers are :
a = 4   b = 9

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to SWAP three variables without using fourth variable.

```
#include <stdio.h>
intmain() {
int a, b, c;
printf ("Enter 1st number: ");
scanf ("%d", &a);
printf ("Enter 2nd number: ");
scanf ("%d", &b);
printf ("Enter 3rd number: ");
scanf ("%d", &c);
 a = a+b; printf("\na=%d",a);
    b = b+c; printf(" b=%d",b);
    c = c+a; printf(" c=%d",c);
    a = c-a; printf(" a=%d",a);
    b = c-b; printf(" b=%d",b);
    c = c-(a+b); printf(" c=%d\n",c);
printf("\nSwap numbers are a = %d, b = %d & c = %d", a, b, c);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

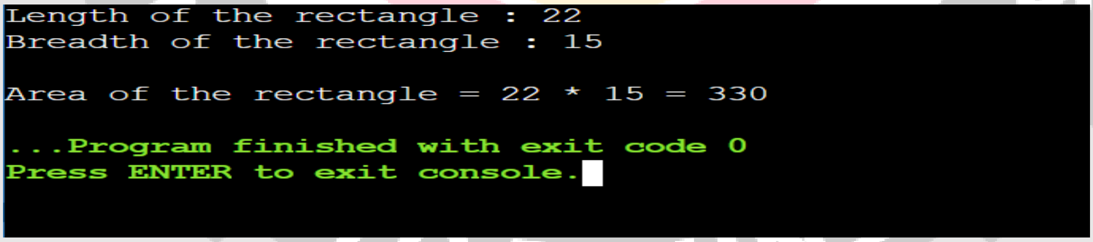


```
Enter 1st number: 10
Enter 2nd number: 20
Enter 3rd number: 30

a=30 b=50 c=60 a=30 b=10 c=20

Swap numbers are a = 30, b = 10 & c = 20

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to find area of rectangle.

```
#include <stdio.h>

Int main() {

int l, b, area;

printf ("Length of the rectangle : ");

scanf ("%d", &l);

printf ("Breadth of the rectangle : ");

scanf ("%d", &b);

area = l * b;

printf ("\nArea of the rectangle = %d * %d = %d", l, b, area);

   return 0;

}

//Vinay Kumar
```

OUTPUT:

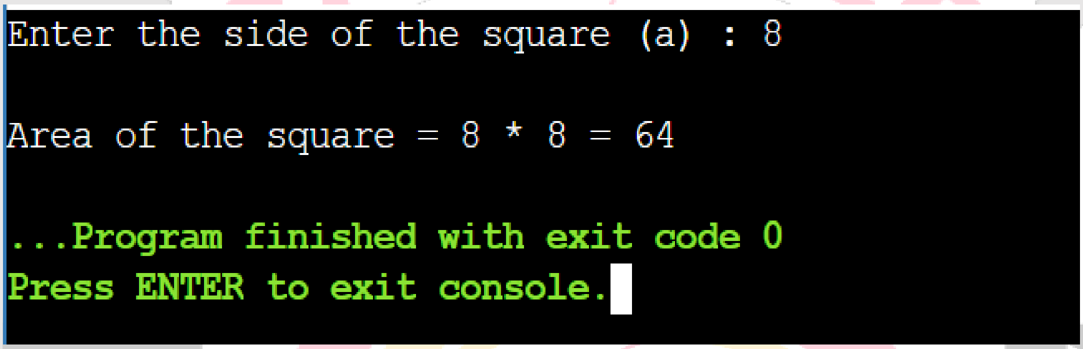


```
Length of the rectangle : 22
Breadth of the rectangle : 15

Area of the rectangle = 22 * 15 = 330

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to find area of square.

```
#include <stdio.h>

intmain() {

int a, area;

printf ("Enter the side of the square (a) : ");
scanf ("%d", &a);
 area = a * a;
printf ("\nArea of the square = %d * %d = %d", a, a, area);
    return 0;
}
//Vinay Kumar
```

OUTPUT:



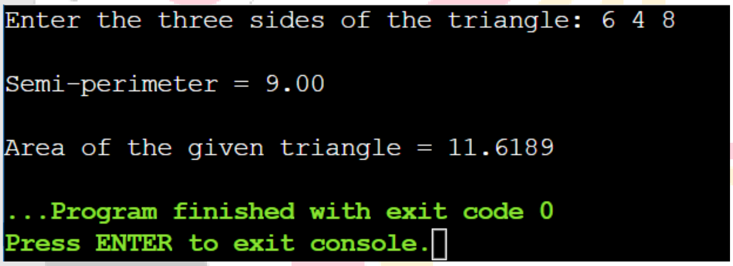
```
Enter the side of the square (a) : 8

Area of the square = 8 * 8 = 64

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to find area of right-angle triangle, isosceles triangle, any triangle with three sides.

<u>AREA OF ISOSCELES TRIANGLE:</u>

```
#include <stdio.h>
#include <math.h>
intmain() {
    float a, b, c, s, area;
printf("Enter the three sides of the triangle: ");
scanf("%f %f %f", &a, &b, &c);
    s = (a+b+c) / 2;
printf("\nSemi-perimeter = %.2f\n", s);
    area = sqrt(s*(s-a)*(s-b)*(s-c));
printf("\nArea of the given triangle = %.4f", area);
    return 0;
}
//Vinay Kumar
```

OUTPUT:



```
Enter the three sides of the triangle: 6 4 8

Semi-perimeter = 9.00

Area of the given triangle = 11.6189

...Program finished with exit code 0
Press ENTER to exit console.
```

AREA OF RIGHT-ANGLED TRIANGLE:

```
#include <stdio.h>
int main() {

int b, h, area;
printf ("Enter the base of the triangle: ");
scanf ("%d", &b);
```

printf ("Enter the height of the triangle: ");

scanf ("%d", &h);

   area = 1/2 * b * h;

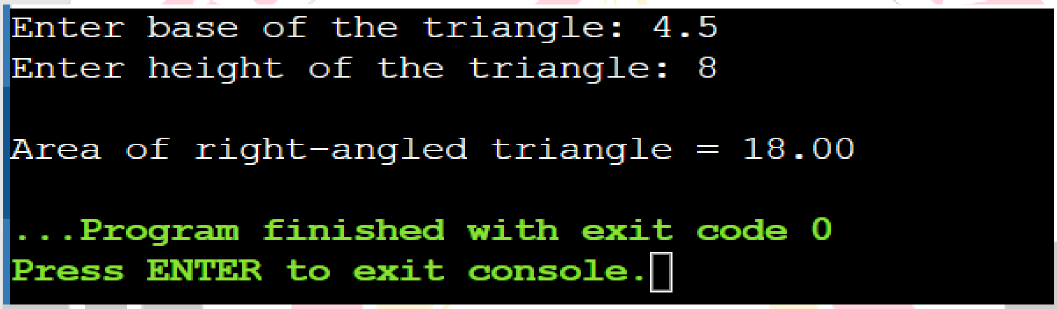
printf ("Area of the triangle = %d", area);

   return 0;

}

//Vinay Kumar

OUTPUT:



```
Enter base of the triangle: 4.5
Enter height of the triangle: 8

Area of right-angled triangle = 18.00

...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to find area and volume of a cube

AREA OF CUBE

#include <stdio.h>

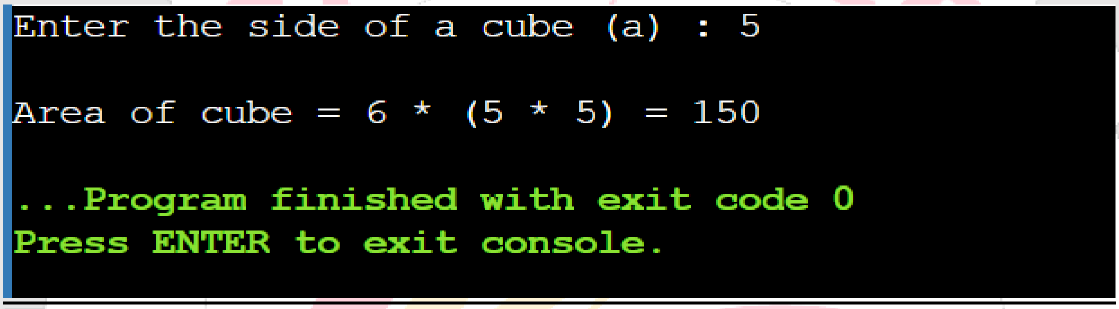
int main() {

int a, area;

printf ("Enter the side of a cube (a) : ");

```
scanf ("%d", &a);
    area = 6 * (a * a);
printf ("\nArea of cube = %d * (%d * %d) = %d", 6, a, a, area);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

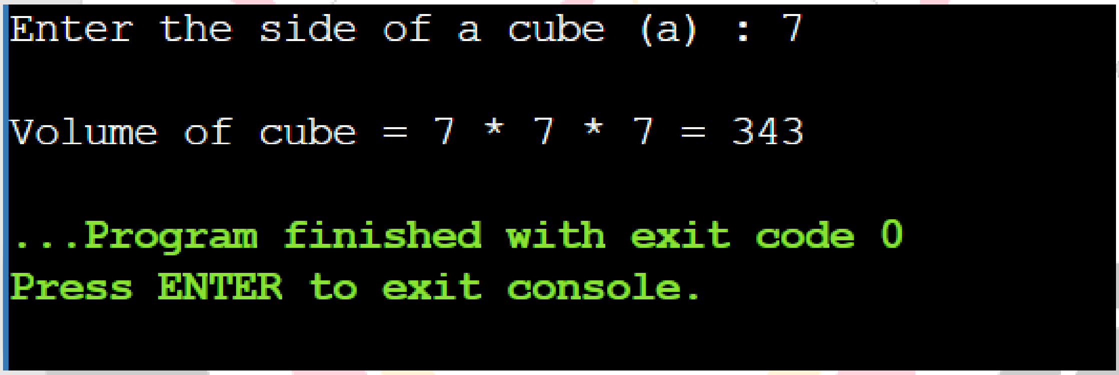


```
Enter the side of a cube (a) : 5

Area of cube = 6 * (5 * 5) = 150

...Program finished with exit code 0
Press ENTER to exit console.
```

VOLUME OF CUBE

```
#include <stdio.h>
int main() {
int a, volume;
printf ("Enter the side of a cube (a) : ");
scanf ("%d", &a);
    volume = a * a * a;
printf ("\nVolume of cube = %d * %d * %d = %d", a, a, a,
volume);
    return 0;
}
//Vinay Kumar
```

OUTPUT:



```
Enter the side of a cube (a) : 7

Volume of cube = 7 * 7 * 7 = 343

...Program finished with exit code 0
Press ENTER to exit console.
```

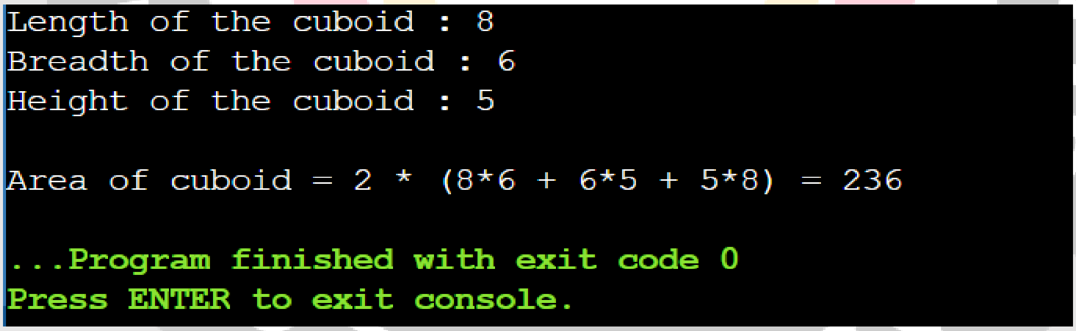
Write a program to find area and volume of a cuboid

AREA OF CUBOID

```
#include <stdio.h>
int main() {
```

```
int l, b, h, area;
printf ("Length of the cuboid : ");
scanf ("%d", &l);
printf ("Breadth of the cuboid : ");
scanf ("%d", &b);
printf ("Height of the cuboid : ");
scanf ("%d", &h);
    area = 2 * (l*b + b*h + h*l);
printf ("\nArea of cuboid = %d * (%d*%d + %d*%d + %d*%d) = %d", 2,l,b,b,h,h,l, area);
    return 0;
}
//Vinay Kumar
```

OUTPUT:



```
Length of the cuboid : 8
Breadth of the cuboid : 6
Height of the cuboid : 5

Area of cuboid = 2 * (8*6 + 6*5 + 5*8) = 236

...Program finished with exit code 0
Press ENTER to exit console.
```

VOLUME OF CUBOID

```
#include <stdio.h>
int main() {
```

```
int l, b, h, volume;
printf ("Length of the cuboid : ");
scanf ("%d", &l);
printf ("Breadth of the cuboid : ");
scanf ("%d", &b);
printf ("Height of the cuboid : ");
scanf ("%d", &h);
    volume = l * b * h;
printf ("\nVolume of cuboid = %d * %d * %d = %d", l, b, h,
volume);
    return 0;
}
//Vinay Kumar
```

OUTPUT:

23. Write a C program units Rs. 0.75/unit

For next 100 units Rs. 1.20/unit

For unit above 250 Rs. 1.50/unit

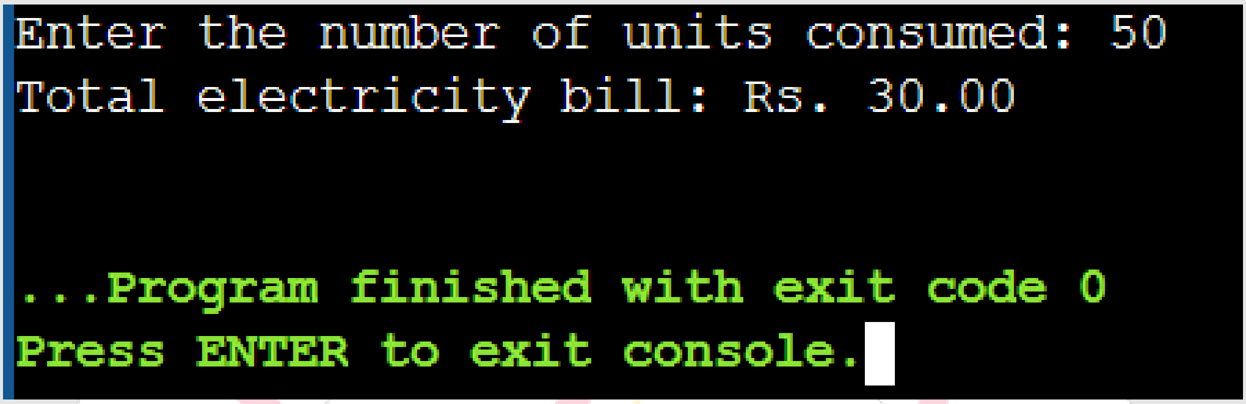
An additional surcharge of 20% is added to the bill

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

```
int main() {
    float units, bill;

    printf("Enter the number of units consumed: ");
    scanf("%f", &units);

    if (units <= 50) {
        bill = units * 0.50;
    } else if (units <= 150) {
        bill = 50 * 0.50 + (units - 50) * 0.75;
    } else if (units <= 250) {
        bill = 50 * 0.50 + 100 * 0.75 + (units - 150) * 1.20;
    } else {
        bill = 50 * 0.50 + 100 * 0.75 + 100 * 1.20 + (units - 250) * 1.50;
    }

    bill += 0.2 * bill;

    printf("Total electricity bill: Rs. %.2f\n", bill);

    return 0;
```

//Vinay Kumar

output:



```
Enter the number of units consumed: 50
Total electricity bill: Rs. 30.00


...Program finished with exit code 0
Press ENTER to exit console.
```

22. Write a C program to input basic salary of an employee and calculate its Gross salary according to following:

Basic Salary <= 10000 : HRA = 20%, DA = 80%

Basic Salary <= 20000 : HRA = 25%, DA = 90%

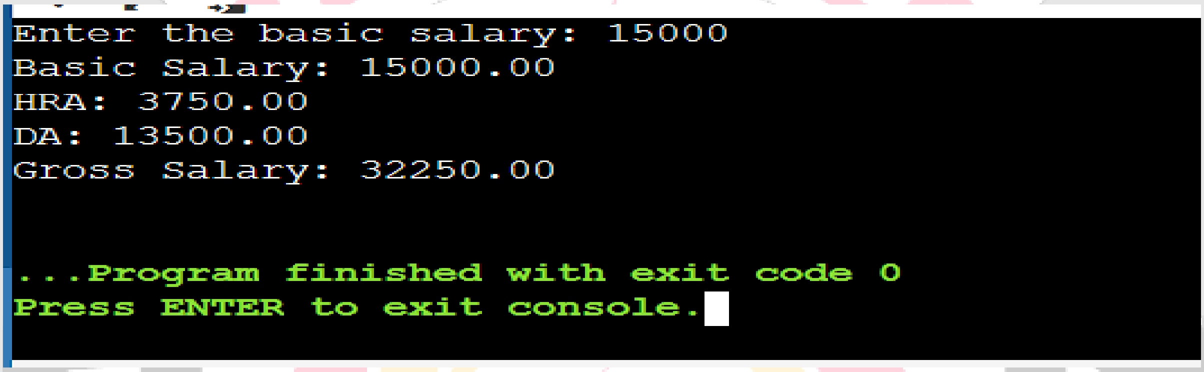
Basic Salary > 20000 : HRA = 30%, DA = 95%

```
#include<stdio.h>
int main() {
   float basic_salary, hra, da, gross_salary;
```

```
// Input basic salary
printf("Enter the basic salary: ");
scanf("%f", &basic_salary);

// Calculate HRA and DA based on conditions
if (basic_salary <= 10000) {
    hra = 0.2 * basic_salary;
    da = 0.8 * basic_salary;
} else if (basic_salary <= 20000) {
    hra = 0.25 * basic_salary;
    da = 0.9 * basic_salary;
} else {
    hra = 0.3 * basic_salary;
    da = 0.95 * basic_salary;
}

// Calculate gross salary
gross_salary = basic_salary + hra + da;

// Print the results
printf("Basic Salary: %.2f\n", basic_salary);
printf("HRA: %.2f\n", hra);
printf("DA: %.2f\n", da);
```

```
    printf("Gross Salary: %.2f\n", gross_salary);


    return 0;
}
//Vinay Kumar
```

Output :



```
Enter the basic salary: 15000
Basic Salary: 15000.00
HRA: 3750.00
DA: 13500.00
Gross Salary: 32250.00


...Program finished with exit code 0
Press ENTER to exit console.
```

21. Write a C program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade according to following:

Percentage >= 90% : Grade A

Percentage >= 80% : Grade B

Percentage >= 70% : Grade C

Percentage >= 60% : Grade D

Percentage >= 40% : Grade E

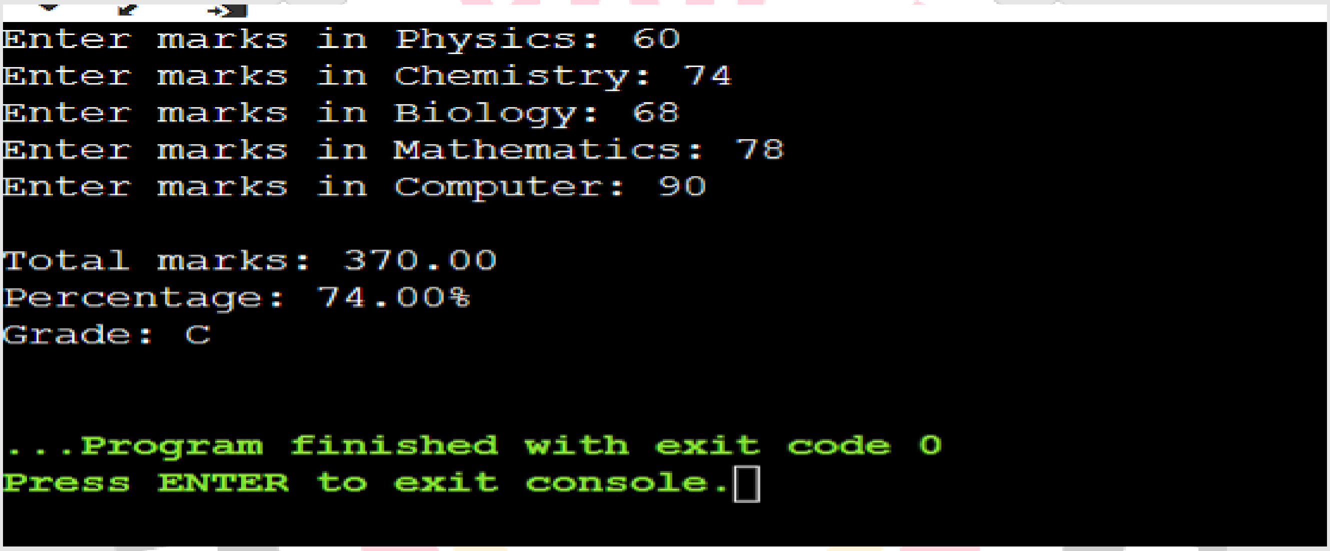
Percentage < 40% : Grade F

```
#include <stdio.h>

int main() {
    float physics, chemistry, biology, mathematics, computer;
    float total_marks, percentage;
    char grade;

    printf("Enter marks in Physics: ");
    scanf("%f", &physics);

    printf("Enter marks in Chemistry: ");
    scanf("%f", &chemistry);

    printf("Enter marks in Biology: ");
    scanf("%f", &biology);

    printf("Enter marks in Mathematics: ");
    scanf("%f", &mathematics);

    printf("Enter marks in Computer: ");
    scanf("%f", &computer);
    total_marks = physics + chemistry + biology + mathematics + computer;

    percentage = (total_marks / 500) * 100;
    if (percentage >= 90) {
        grade = 'A';
    } else if (percentage >= 80) {
        grade = 'B';
    } else if (percentage >= 70) {
        grade = 'C';
    } else if (percentage >= 60) {
        grade = 'D';
    } else if (percentage >= 40) {
        grade = 'E';
    } else {
        grade = 'F';
    }
    printf("\nTotal marks: %.2f\n", total_marks);
    printf("Percentage: %.2f%%\n", percentage);
    printf("Grade: %c\n", grade);

    return 0;
```

```
}
```

//Vinay Kumar

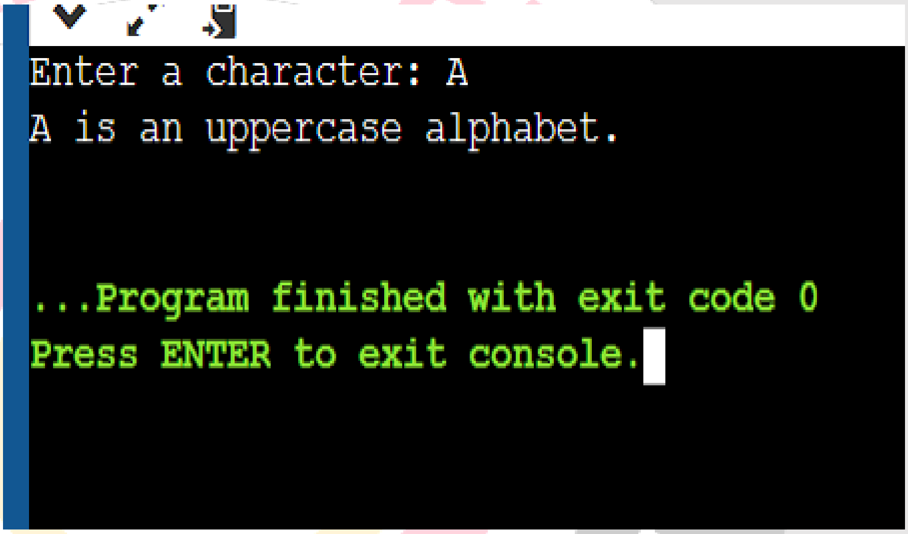
Output :



```
Enter marks in Physics: 60
Enter marks in Chemistry: 74
Enter marks in Biology: 68
Enter marks in Mathematics: 78
Enter marks in Computer: 90

Total marks: 370.00
Percentage: 74.00%
Grade: C


...Program finished with exit code 0
Press ENTER to exit console.
```

10. Write a C program to check whether a character is uppercase or lowercase alphabet.

```
#include <stdio.h>
int main() {
    char c;
    printf("Enter a character: ");
    scanf("%c", &c);
   if (c >= 'A' && c <= 'Z') {
        printf("%c is an uppercase alphabet.\n", c);
    }
    else if (c >= 'a' && c <= 'z') {
        printf("%c is a lowercase alphabet.\n", c);
    }
    else {
        printf("%c is not an alphabet.\n", c);
    }


    return 0;
    //Vinay Kumar
}
```

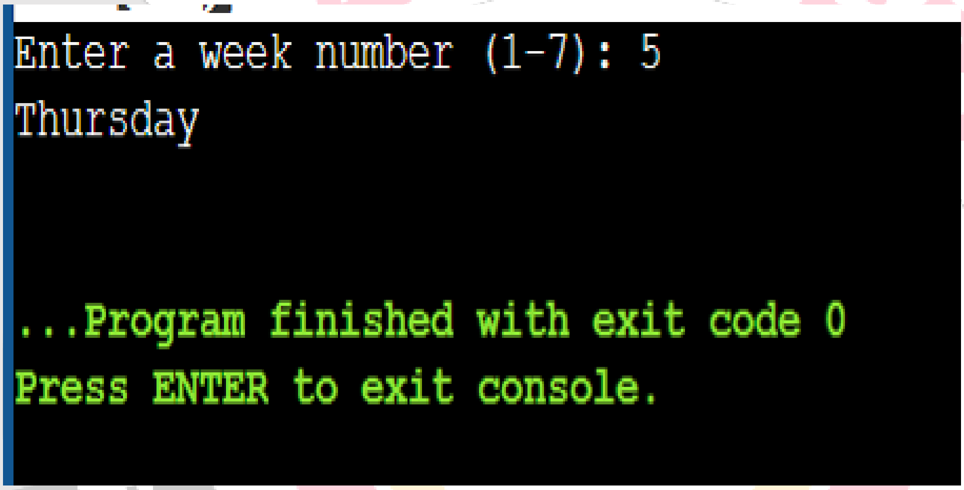


```
Enter a character: A
A is an uppercase alphabet.


...Program finished with exit code 0
Press ENTER to exit console.
```

11. Write a C program to input week number and print week day.

```
#include <stdio.h>


int main() {
    int weekNumber;
    printf("Enter a week number (1-7): ");
    scanf("%d", &weekNumber);
    switch(weekNumber) {
        case 1:
            printf("Sunday\n");
            break;
        case 2:
            printf("Monday\n");
            break;
        case 3:
            printf("Tuesday\n");
            break;
        case 4:
            printf("Wednesday\n");
            break;
        case 5:
            printf("Thursday\n");
            break;
        case 6:
            printf("Friday\n");
            break;
        case 7:
            printf("Saturday\n");
            break;
        default:
```

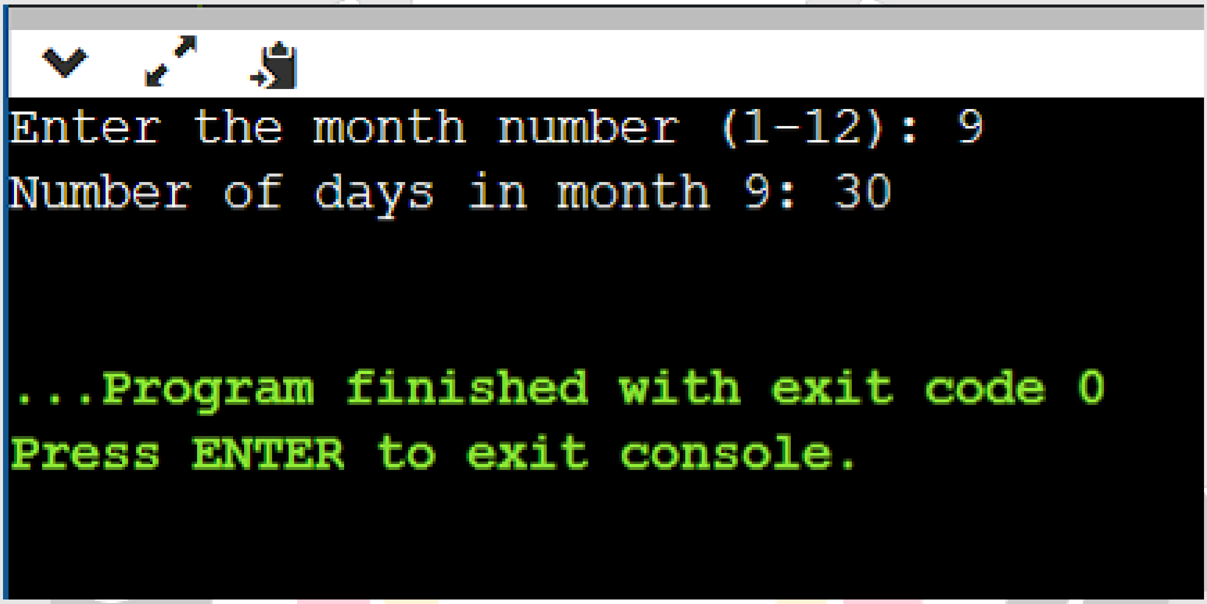
```
        printf("Invalid input. Please enter a number between 1 and 7.\n");
    }

    return 0;
}
//Vinay Kumar
```



```
Enter a week number (1-7): 5
Thursday


...Program finished with exit code 0
Press ENTER to exit console.
```

12. Write a C program to input month number and print number of days in that month.

```
#include <stdio.h>

int main() {
    int month;

    printf("Enter the month number (1-12): ");
    scanf("%d", &month);


    if (month < 1 || month > 12) {
        printf("Invalid month number. Please enter a number between 1 and 12.\n");
        return 1; // Return an error code
    }
    int days;
    switch(month) {
        case 2: // February
            days = 28;
            break;
        case 4: case 6: case 9: case 11: // April, June, September, November
            days = 30;
            break;
        default: // All other months
            days = 31;
            break;
    }



    printf("Number of days in month %d: %d\n", month, days);
    return 0;
}
//Vinay Kumar
```

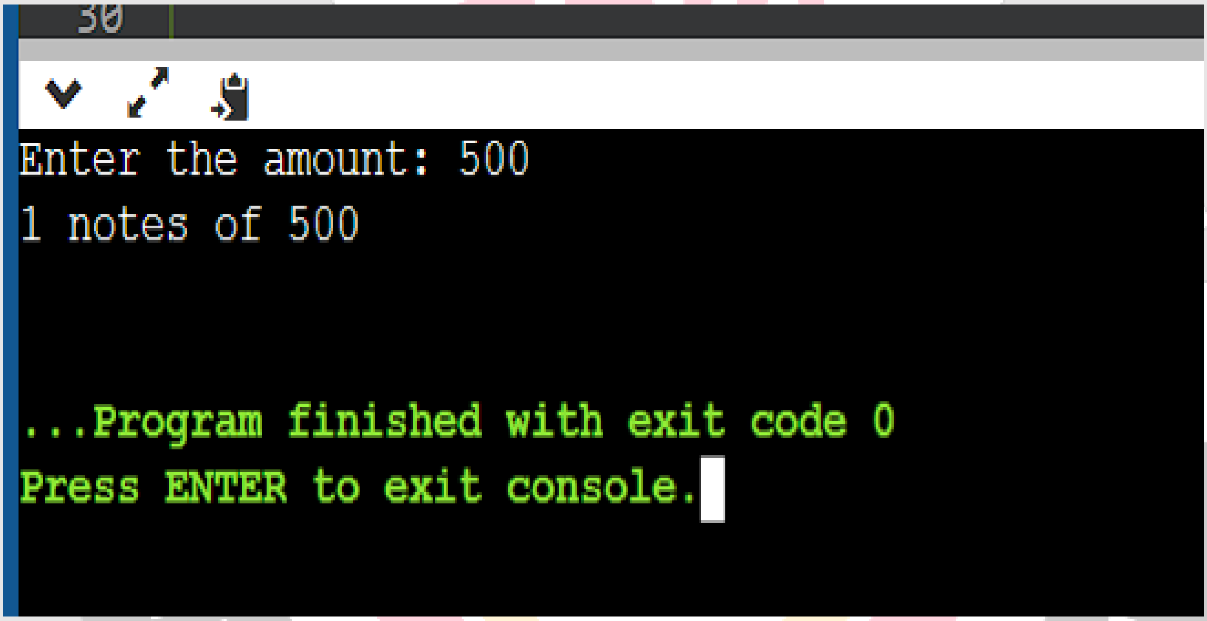


```
Enter the month number (1-12): 9
Number of days in month 9: 30


...Program finished with exit code 0
Press ENTER to exit console.
```

13. Write a C program to count total number of notes in given amount.

```
#include <stdio.h>

int main() {
    int amount;
    int notes[6] = {2000, 500, 100, 50, 20, 10}; //
Denominations of notes
```

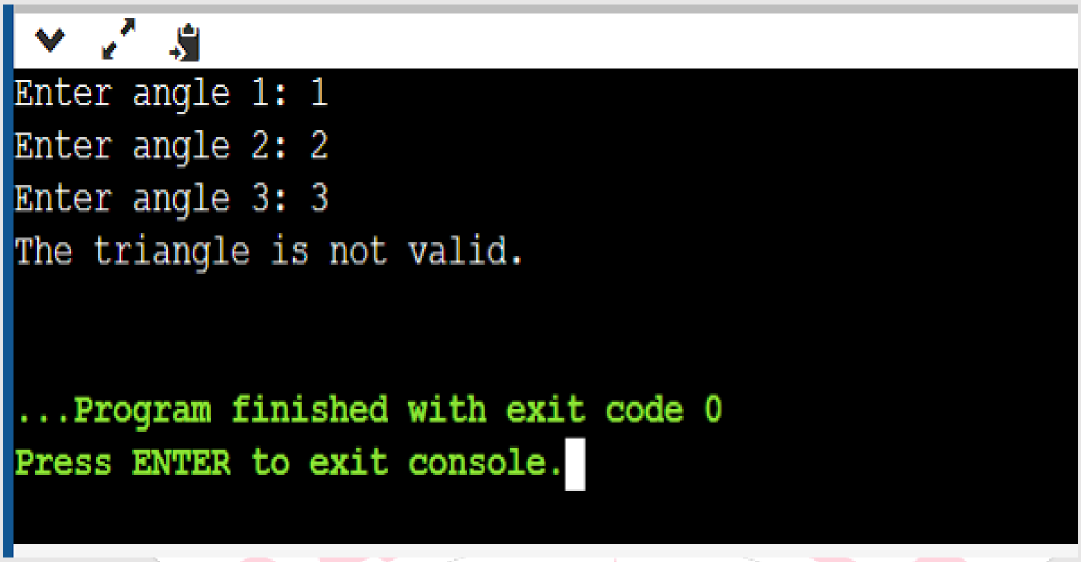
```
    // Input the amount
    printf("Enter the amount: ");
    scanf("%d", &amount);

    // Initialize a counter for each type of note
    int note_count[6] = {0};

    // Iterate through the notes and count how many of each are needed
    for (int i = 0; i < 6; i++) {
        note_count[i] = amount / notes[i];
        amount = amount % notes[i];
    }

    // Display the results
    for (int i = 0; i < 6; i++) {
        if (note_count[i] > 0) {
            printf("%d notes of %d\n", note_count[i], notes[i]);
        }
    }
    return 0;
}
//Vinay Kumar
```



```
Enter the amount: 500
1 notes of 500


...Program finished with exit code 0
Press ENTER to exit console.
```

16. Write a C program to input angles of a triangle and check whether triangle is valid or not.

```
#include <stdio.h>

int main() {
    int angle1, angle2, angle3;
    printf("Enter angle 1: ");
    scanf("%d", &angle1);
printf("Enter angle 2: ");
    scanf("%d", &angle2);

    printf("Enter angle 3: ");
    scanf("%d", &angle3);

if (angle1 + angle2 + angle3 == 180 && angle1 > 0 && angle2 > 0 && angle3 > 0) {
        printf("The triangle is valid.\n");
    } else {
        printf("The triangle is not valid.\n");
    }

    return 0;
}
//Vinay Kumar
```

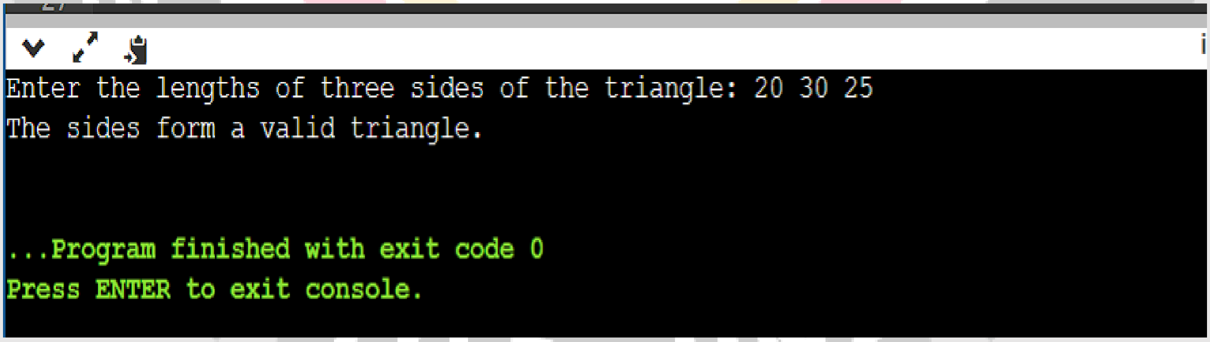


```
Enter angle 1: 1
Enter angle 2: 2
Enter angle 3: 3
The triangle is not valid.


...Program finished with exit code 0
Press ENTER to exit console.
```

17. Write a C program to input all sides of a triangle and check whether triangle is valid or not.

```
#include <stdio.h>
int main() {
   float side1, side2, side3;
```

```
 printf("Enter the lengths of three sides of the
triangle: ");
    scanf("%f %f %f", &side1, &side2, &side3);


    if (side1 + side2 > side3 && side1 + side3 > side2
&& side2 + side3 > side1) {
        printf("The sides form a valid triangle.\n");
    } else {
        printf("The sides do not form a valid
triangle.\n");
    }


    return 0;
}
```



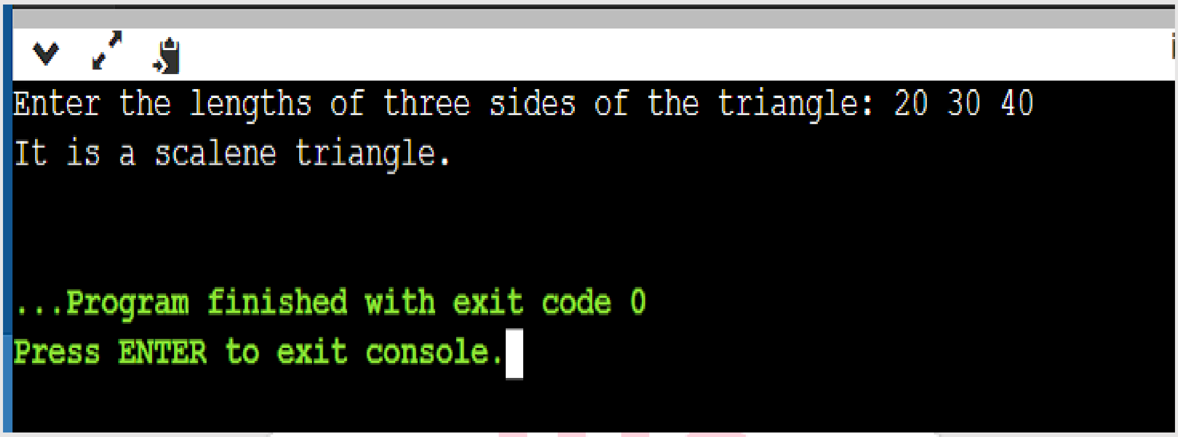
```
Enter the lengths of three sides of the triangle: 20 30 25
The sides form a valid triangle.


...Program finished with exit code 0
Press ENTER to exit console.
```

//Vinay Kumar

18. Write a C program to check whether the triangle is equilateral, isosceles or scalene triangle.

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

```
int main() {
    float side1, side2, side3;
    printf("Enter the lengths of three sides of the
triangle: ");
    scanf("%f %f %f", &side1, &side2, &side3);
    if (side1 + side2 > side3 && side1 + side3 > side2
&& side2 + side3 > side1) {
        if (side1 == side2 && side2 == side3) {
            printf("It is an equilateral triangle.\n");
        } else if (side1 == side2 || side1 == side3 || side2
== side3) {
            printf("It is an isosceles triangle.\n");
        } else {
            printf("It is a scalene triangle.\n");
        }
    } else {
        printf("The given sides do not form a valid
triangle.\n");
    }

    return 0;
}
//Vinay Kumar
```

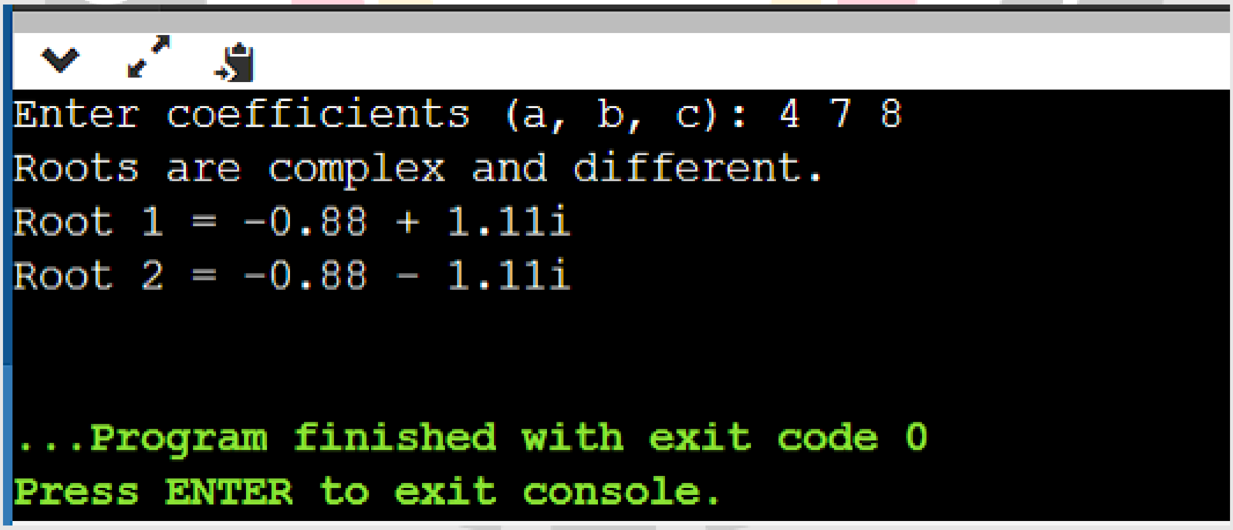


```
Enter the lengths of three sides of the triangle: 20 30 40
It is a scalene triangle.


...Program finished with exit code 0
Press ENTER to exit console.
```

19. Write a C program to find all roots of a quadratic equation.

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

```
#include <math.h>

int main() {
    double a, b, c;
    double discriminant, root1, root2;

    printf("Enter coefficients (a, b, c): ");
    scanf("%lf %lf %lf", &a, &b, &c);
    discriminant = b*b - 4*a*c;

    if (discriminant > 0) {
        root1 = (-b + sqrt(discriminant)) / (2*a);
        root2 = (-b - sqrt(discriminant)) / (2*a);
        printf("Roots are real and different.\n");
        printf("Root 1 = %.2lf\n", root1);
        printf("Root 2 = %.2lf\n", root2);
    }
    else if (discriminant == 0) {
        root1 = -b / (2*a);
        printf("Roots are real and same.\n");
        printf("Root 1 = Root 2 = %.2lf\n", root1);
    }
    else {
        double realPart = -b / (2*a);
        double imaginaryPart = sqrt(-discriminant) /
(2*a);
        printf("Roots are complex and different.\n");
        printf("Root 1 = %.2lf + %.2lfi\n", realPart,
imaginaryPart);
        printf("Root 2 = %.2lf - %.2lfi\n", realPart,
imaginaryPart);
    }

    return 0;
}
//Vinay Kumar
```



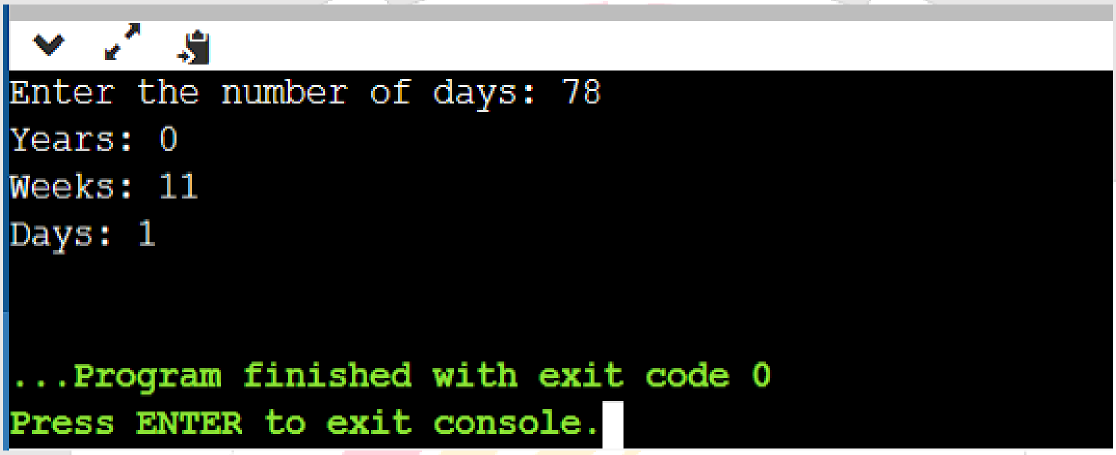
```
Enter coefficients (a, b, c): 4 7 8
Roots are complex and different.
Root 1 = -0.88 + 1.11i
Root 2 = -0.88 - 1.11i


...Program finished with exit code 0
Press ENTER to exit console.
```

.24. Write a C program to convert specified days into years, weeks and days.

```
#include <stdio.h>
int main()
```

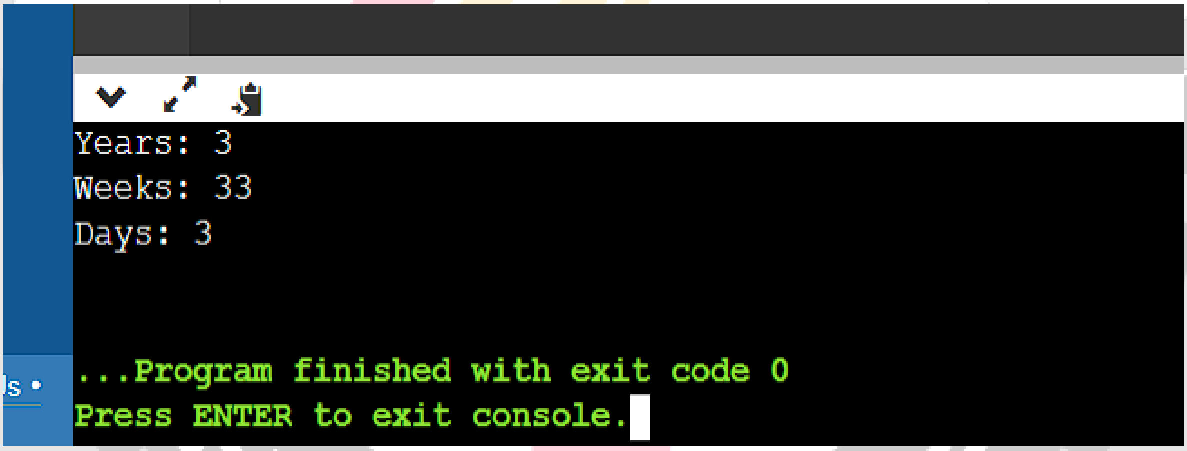
```
{
 int days, years, weeks;
 days = 1329;

#include <stdio.h>

int main() {
   int days, years, weeks;

   // Input the number of days
   printf("Enter the number of days: ");
   scanf("%d", &days);

   // Convert days into years, weeks, and days
   years = days / 365;
   days = days % 365;
   weeks = days / 7;
   days = days % 7;

   // Output the result
   printf("Years: %d\n", years);
   printf("Weeks: %d\n", weeks);
   printf("Days: %d\n", days);
```

```
    return 0;
}
//Vinay Kumar
```



```
Enter the number of days: 78
Years: 0
Weeks: 11
Days: 1


...Program finished with exit code 0
Press ENTER to exit console.
```

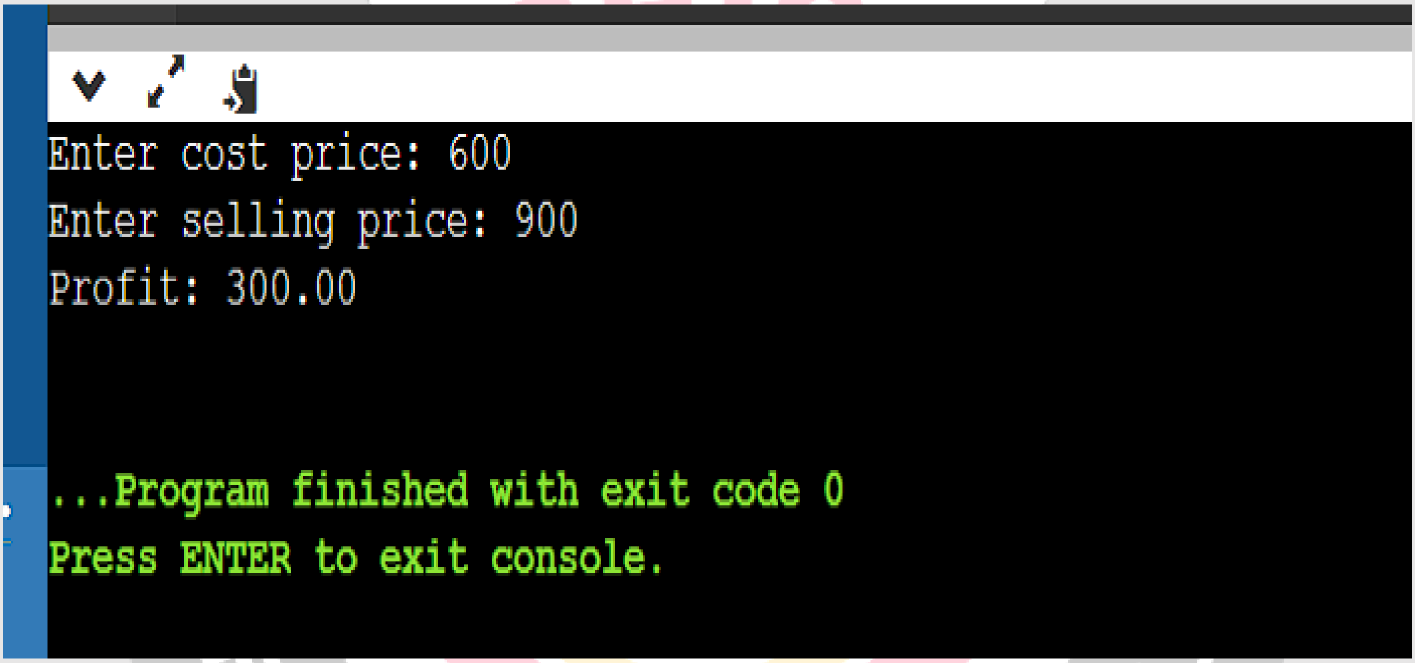
```
#include <stdio.h>

int main() {
    int days, years, weeks;
    days = 1329;

    // Converts days to years, weeks, and days
    years = days / 365;
    int remainingDays = days % 365;

    weeks = remainingDays / 7;
    int remainingDays2 = remainingDays % 7;
    printf("Years: %d\n", years);
    printf("Weeks: %d\n", weeks);
    printf("Days: %d\n", remainingDays2);

    return 0;
}
//Vinay Kumar
```



```
Years: 3
Weeks: 33
Days: 3


...Program finished with exit code 0
Press ENTER to exit console.
```

20. Write a C program to calculate profit or loss.

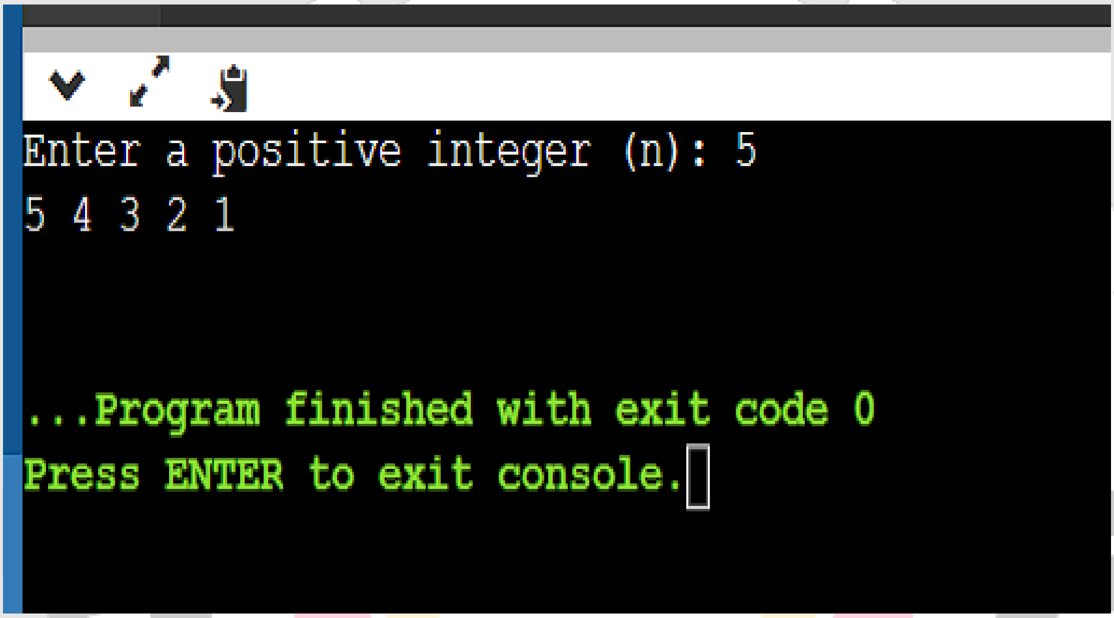
```
#include <stdio.h>

int main() {
    float cost_price, selling_price, profit_loss;

    // Get cost price and selling price from the user
    printf("Enter cost price: ");
    scanf("%f", &cost_price);

    printf("Enter selling price: ");
    scanf("%f", &selling_price);

    // Calculate profit or loss
    profit_loss = selling_price - cost_price;

    // Determine if it's a profit or loss and display the result
    if (profit_loss > 0) {
        printf("Profit: %.2f\n", profit_loss);
    } else if (profit_loss < 0) {
        printf("Loss: %.2f\n", -profit_loss);
    } else {
        printf("No profit, no loss.\n");
    }

    return 0;
}
//Vinay Kumar
```



```
Enter cost price: 600
Enter selling price: 900
Profit: 300.00


...Program finished with exit code 0
Press ENTER to exit console.
```

23. Write a C program to print all natural numbers in reverse (from n to 1).

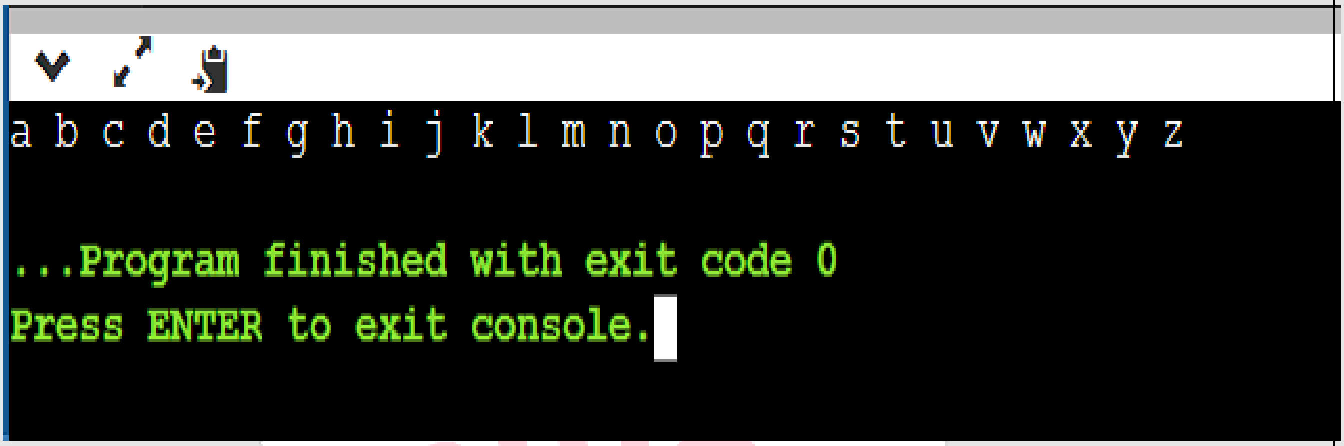
```
#include <stdio.h>

int main() {
    int n;

    // Prompt user for input
    printf("Enter a positive integer (n): ");
    scanf("%d", &n);

    // Check if n is positive
    if (n <= 0) {
        printf("Please enter a positive integer.\n");
        return 1; // Return an error code
    }

    // Print natural numbers in reverse order
    for (int i = n; i >= 1; i--) {
        printf("%d ", i);
    }
    printf("\n");

    return 0;
}
//Vinay Kumar
```



```
Enter a positive integer (n): 5
5 4 3 2 1


...Program finished with exit code 0
Press ENTER to exit console.
```

24. Write a C program to print all alphabets from a to z.

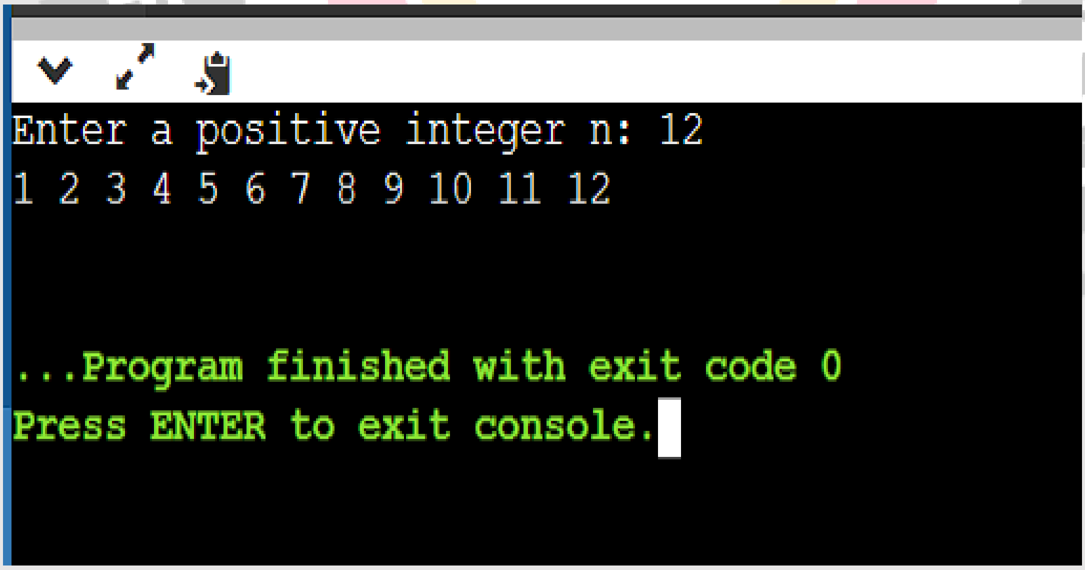
```
#include <stdio.h>

int main() {
    char alphabet;

    for (alphabet = 'a'; alphabet <= 'z'; alphabet++) {
        printf("%c ", alphabet);
    }

    return 0;
}
//Vinay Kumar
```



```
a b c d e f g h i j k l m n o p q r s t u v w x y z

...Program finished with exit code 0
Press ENTER to exit console.
```

25. Write a CWrite a C program to print all natural numbers from 1 to n.

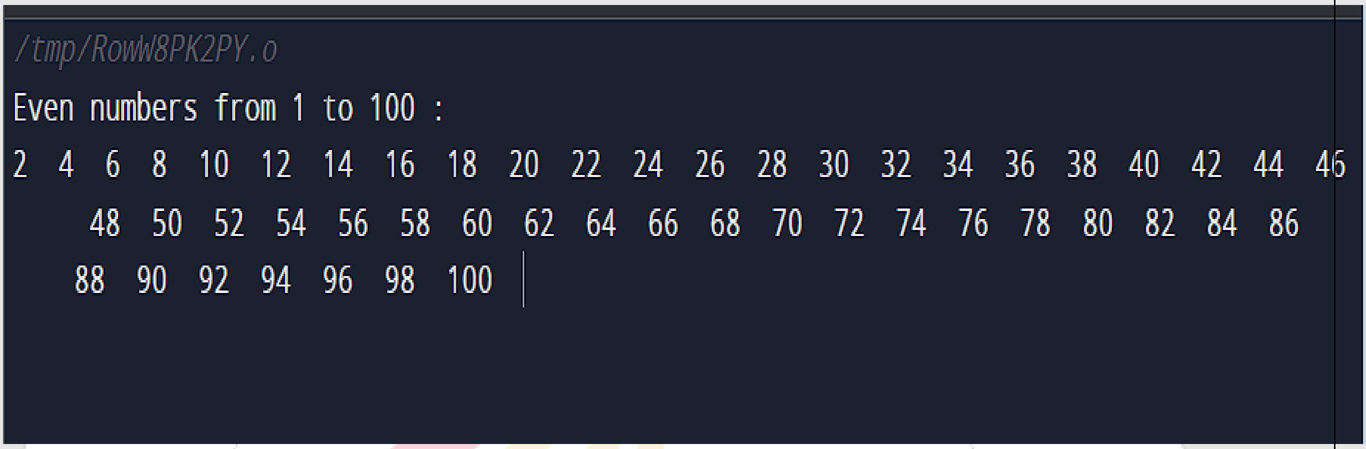
```
#include <stdio.h>

int main() {
    int n, i;

    // Ask the user for the value of n
    printf("Enter a positive integer n: ");
    scanf("%d", &n);

    // Check if n is a positive integer
    if (n <= 0) {
        printf("Please enter a positive integer.\n");
        return 1; // Return an error code
    }

    // Loop from 1 to n and print the natural numbers
    for (i = 1; i <= n; i++) {
        printf("%d ", i);
    }

    printf("\n");

    return 0;
}
//Vinay Kumar
```



```
Enter a positive integer n: 12
1 2 3 4 5 6 7 8 9 10 11 12


...Program finished with exit code 0
Press ENTER to exit console.
```

26. Write a program to print all even numbers between 1 to 100.

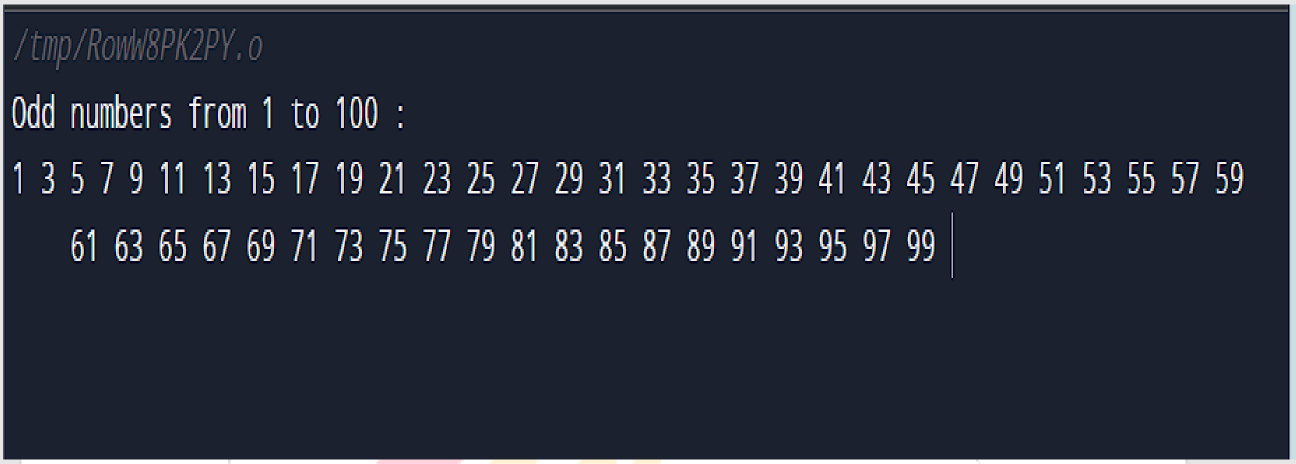
```
#include<stdio.h>
int main() {
  int i;

  printf("\nEven numbers from 1 to 100 :\n");
for (i = 1; i <= 100; i++) {
    if (i % 2 != 0) {
        continue;
    }
    printf("%d  ", i);
  }
  return 0;
}
//Vinay Kumar
```



```
/tmp/RowW8PK2PY.o
Even numbers from 1 to 100 :
2  4  6  8  10  12  14  16  18  20  22  24  26  28  30  32  34  36  38  40  42  44  46
     48  50  52  54  56  58  60  62  64  66  68  70  72  74  76  78  80  82  84  86
    88  90  92  94  96  98  100
```

27. Write a C program to print all odd number between 1 to 100.

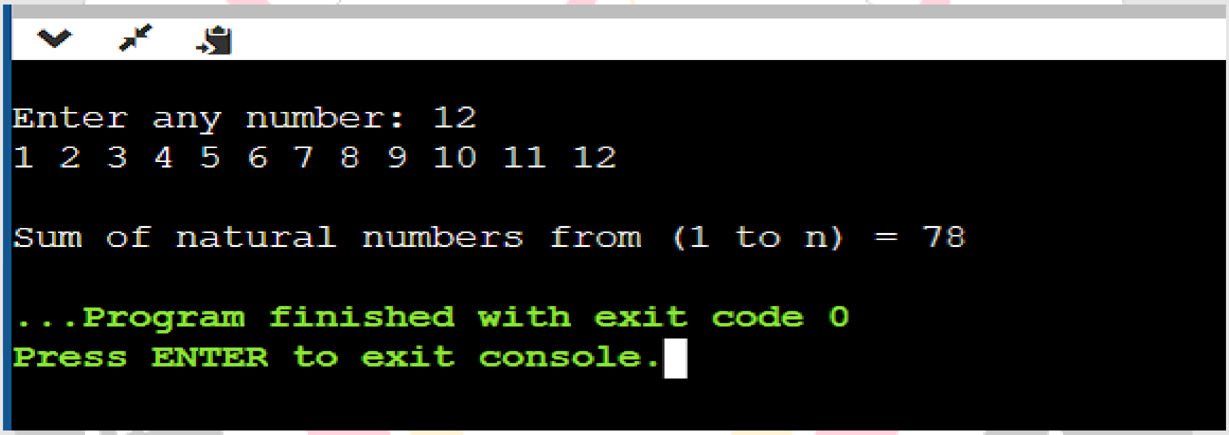
```
#include<stdio.h>
int main() {
  int i;
  printf("\nOdd numbers from 1 to 100 :\n");
for (i = 1; i <= 100; i++) {
    if (i % 2 == 0) {
       continue;
    }
    printf("%d ", i);
  }
  return 0;
}
//Vinay Kumar
```



```
/tmp/RowW8PK2PY.o
Odd numbers from 1 to 100 :
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59
  61 63 65 67 69 71 73 75 77 79 81 83 85 87 89 91 93 95 97 99
```

28. Write a C program to find sum of all natural numbers between 1 to n.

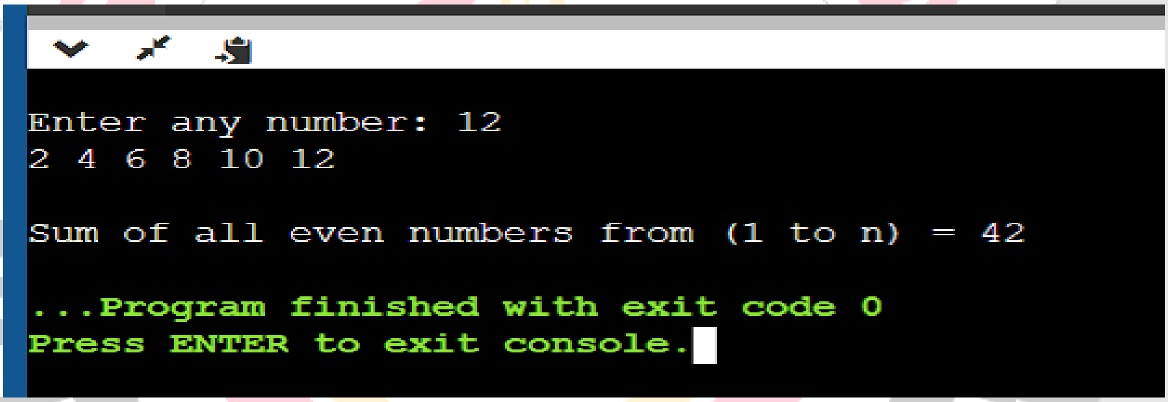
```
#include<stdio.h>
int main()
{
  int i, n, sum = 0;
  printf("\nEnter any number: ");
  scanf("%d", &n);

  for (i = 1; i <= n; i++) {
    printf("%d ", i);
    sum = sum + i;
  }
  printf("\n\nSum of natural numbers from (1 to n) = %d", sum);
  return 0;
}
//Vinay Kumar
```



```
Enter any number: 12
1 2 3 4 5 6 7 8 9 10 11 12

Sum of natural numbers from (1 to n) = 78

...Program finished with exit code 0
Press ENTER to exit console.
```

29. Write a C program to find sum of all even numbers between 1 to n.

```
#include<stdio.h>
int main()
{
  int i, n, sum = 0;
  printf("\nEnter any number: ");
  scanf("%d", &n);
  for (i = 1; i <= n; i++) {
    if (i % 2 != 0) {
        continue;
      }
      printf("%d ", i);
      sum = sum + i;
     }
   printf("\n\nSum of all even numbers from (1 to n) =
%d", sum);
   return 0;
}
```

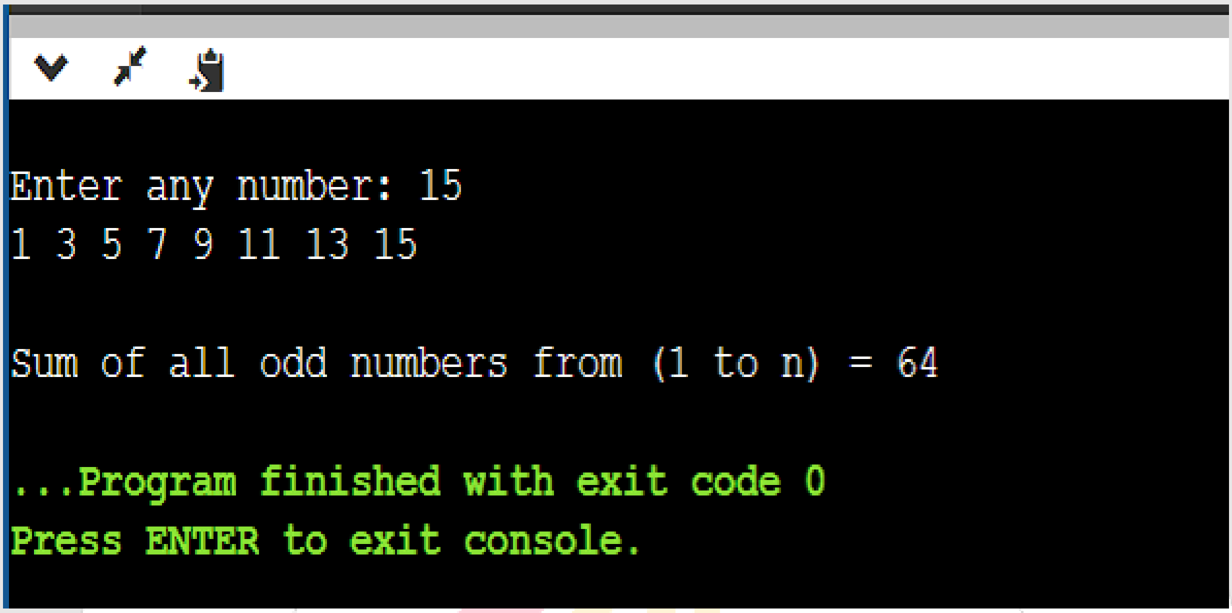


```
Enter any number: 12
2 4 6 8 10 12

Sum of all even numbers from (1 to n) = 42

...Program finished with exit code 0
Press ENTER to exit console.
```

//Vinay Kumar

30.Write a C program to find sum of all odd numbers between 1 to n.

```
#include<stdio.h>
int main()
{
  int i, n, sum = 0;
  printf("\nEnter any number: ");
  scanf("%d", &n);

  for (i = 1; i <= n; i++) {
    if (i % 2 == 0) {
      continue;
    }
    printf("%d ", i);
    sum = sum + i;

  }
  printf("\n\nSum of all odd numbers from (1 to n) =
%d", sum);
  return 0;
}
//Vinay Kumar
```

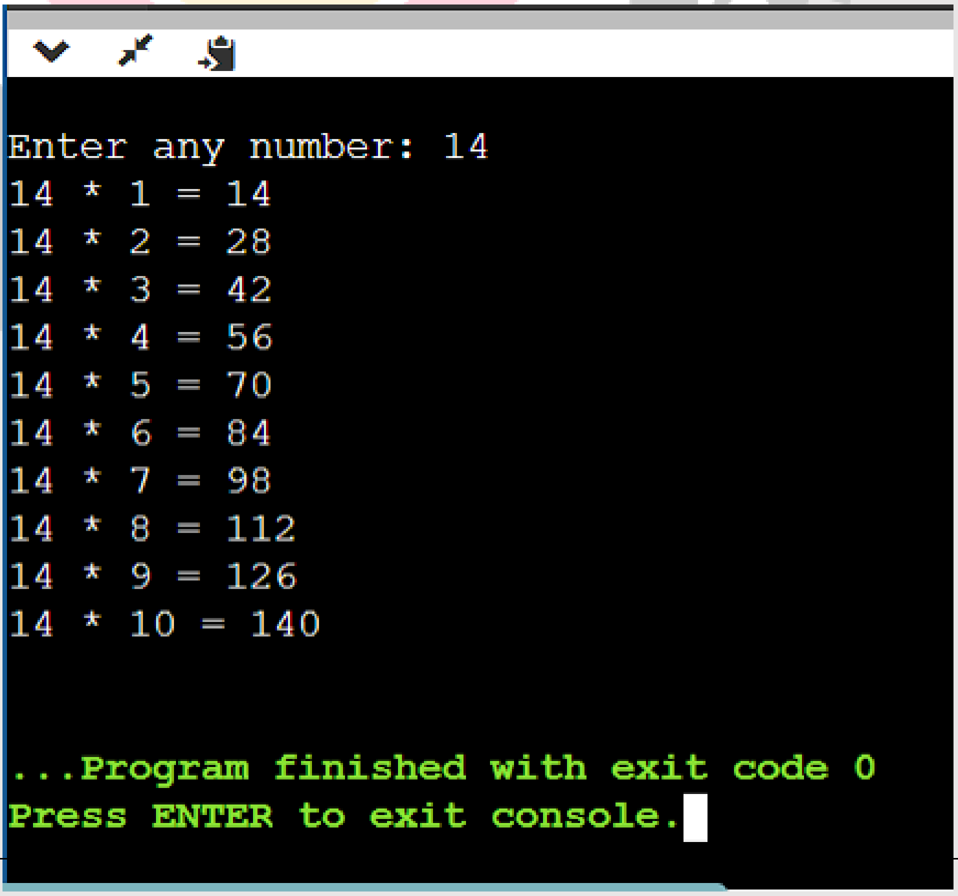


```
Enter any number: 15
1 3 5 7 9 11 13 15

Sum of all odd numbers from (1 to n) = 64

...Program finished with exit code 0
Press ENTER to exit console.
```

31. Write a C program to print multiplication table of any number.

```
#include<stdio.h>
int main()
{
  int i, n, product;
  printf("\nEnter any number: ");
  scanf("%d", &n);

  for (i = 1; i <= 10; i++) {
    product = n * i;
    printf("%d * %d = %d\n", n, i, product);
  }
  return 0;
}
```

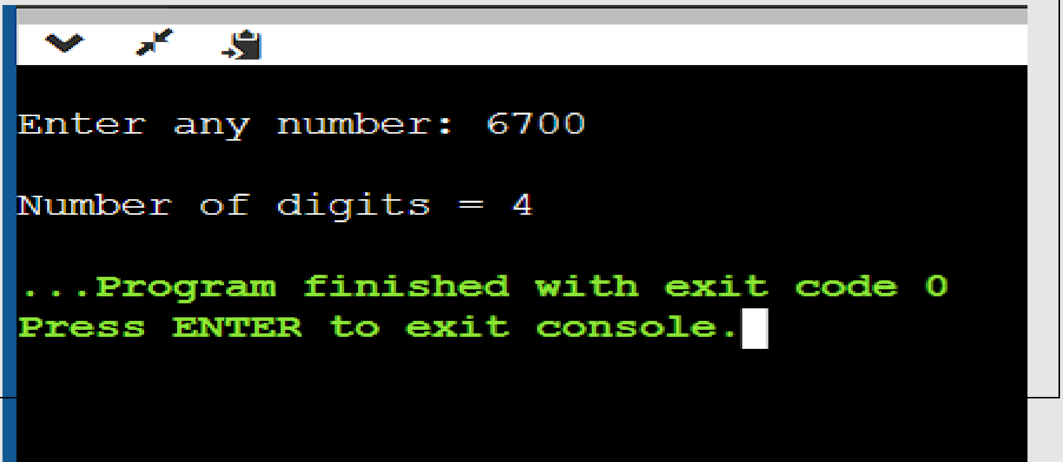
//Vinay Kumar



```
Enter any number: 14
14 * 1 = 14
14 * 2 = 28
14 * 3 = 42
14 * 4 = 56
14 * 5 = 70
14 * 6 = 84
14 * 7 = 98
14 * 8 = 112
14 * 9 = 126
14 * 10 = 140


...Program finished with exit code 0
Press ENTER to exit console.
```

32. Write a C program to count number of digits in a number.

```
#include<stdio.h>
int main()
{

  int n, count = 0;
  printf("\nEnter any number: ");
  scanf("%d", &n);

  while (n > 0) {
    n = n / 10;
    count++;
  }

  printf("\nNumber of digits = %d", count);
  return 0;
}
```

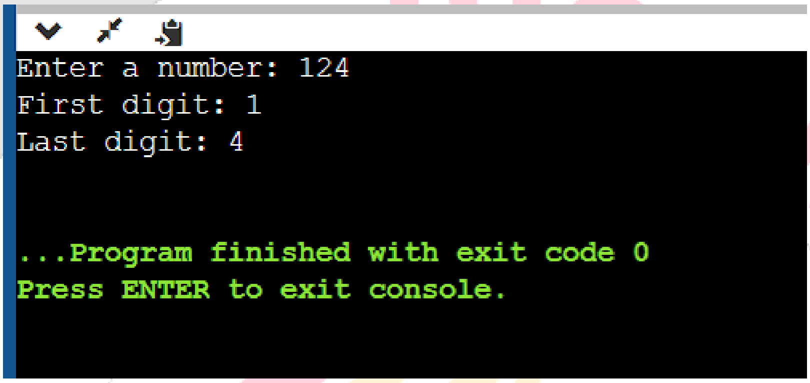


```
Enter any number: 6700

Number of digits = 4

...Program finished with exit code 0
Press ENTER to exit console.
```

```
//Vinay Kumar
```

33. Write a C program to find first and last digit of a number

```
#include <stdio.h>

int main() {
    int number, firstDigit, lastDigit;

    printf("Enter a number: ");
    scanf("%d", &number);
    lastDigit = number % 10;
    while (number >= 10) {
        number /= 10;
    }
    firstDigit = number;

    printf("First digit: %d\n", firstDigit);
    printf("Last digit: %d\n", lastDigit);


    return 0;
}
//Vinay Kumar
```

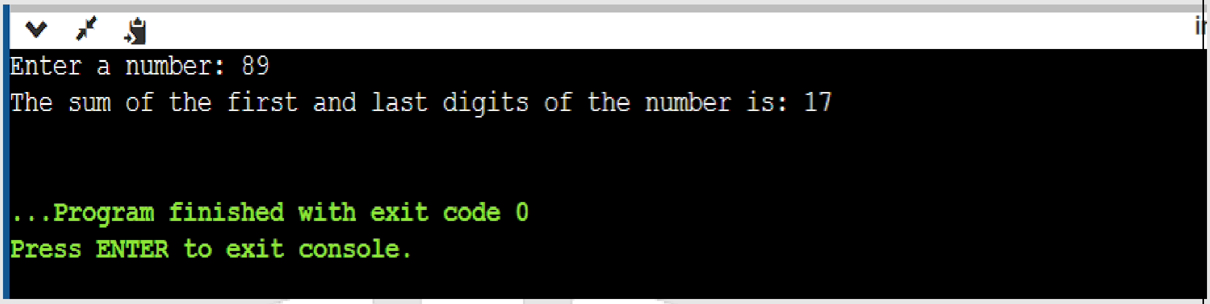


```
Enter a number: 124
First digit: 1
Last digit: 4


...Program finished with exit code 0
Press ENTER to exit console.
```

34. Write a C program to find sum of first and last digit of a number

```
#include <stdio.h>


int main() {
   int num, firstDigit, lastDigit, sum;
   printf("Enter a number: ");
   scanf("%d", &num);
   lastDigit = num % 10;
   while (num >= 10) {
      num /= 10;
   }
   firstDigit = num;
   sum = firstDigit + lastDigit;


   printf("The sum of the first and last digits of the number is: %d\n", sum);
```

```
    return 0;
}
```



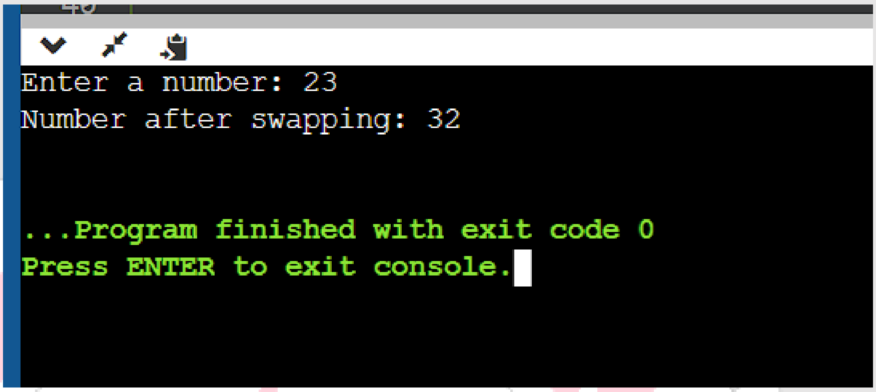
```
Enter a number: 89
The sum of the first and last digits of the number is: 17


...Program finished with exit code 0
Press ENTER to exit console.
```

//Vinay Kumar

35. Write a C program to swap first and last digits of a number.

```
#include <stdio.h>

int main() {
    int number, originalNumber, firstDigit, lastDigit, temp;

    printf("Enter a number: ");
    scanf("%d", &number);

    originalNumber = number;

    // Get the last digit
    lastDigit = number % 10;
    // Find the number of digits in the given number
    while (number >= 10) {
        number /= 10;
    }

    // The first digit is now the remaining number
    firstDigit = number;

    // Swap the first and last digits
    temp = firstDigit;
    firstDigit = lastDigit;
    lastDigit = temp;

    // Reconstruct the new number
    int swappedNumber = 0;
    number = originalNumber;
    while (number >= 10) {
        swappedNumber = swappedNumber * 10 +
(number % 10);
        number /= 10;
    }
    swappedNumber = swappedNumber * 10 +
number;
```

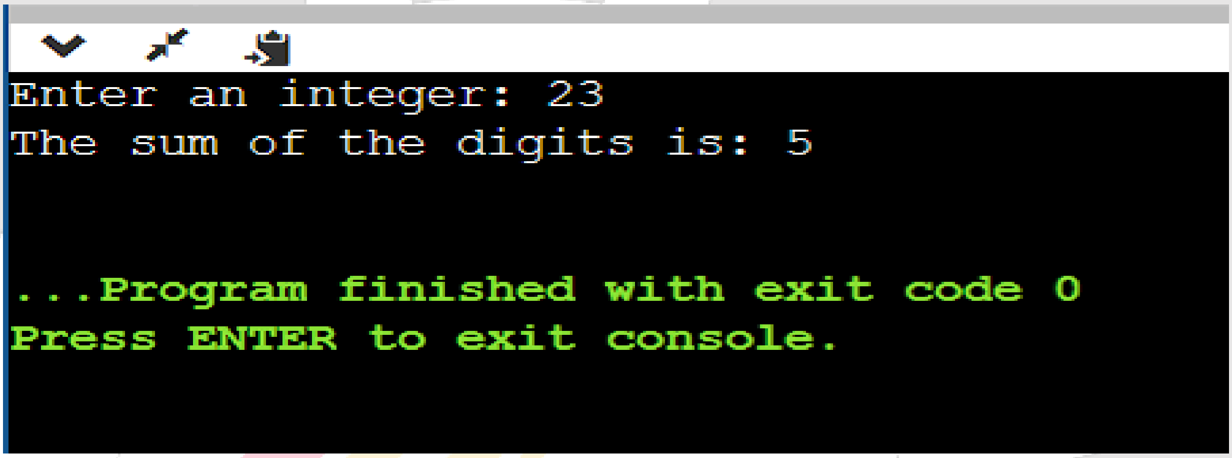
```
    printf("Number after swapping: %d\n",
swappedNumber);

    return 0;
}
//Vinay
Kumar
```



```
Enter a number: 23
Number after swapping: 32


...Program finished with exit code 0
Press ENTER to exit console.
```

36. Write a C program to calculate sum of digits of a number.

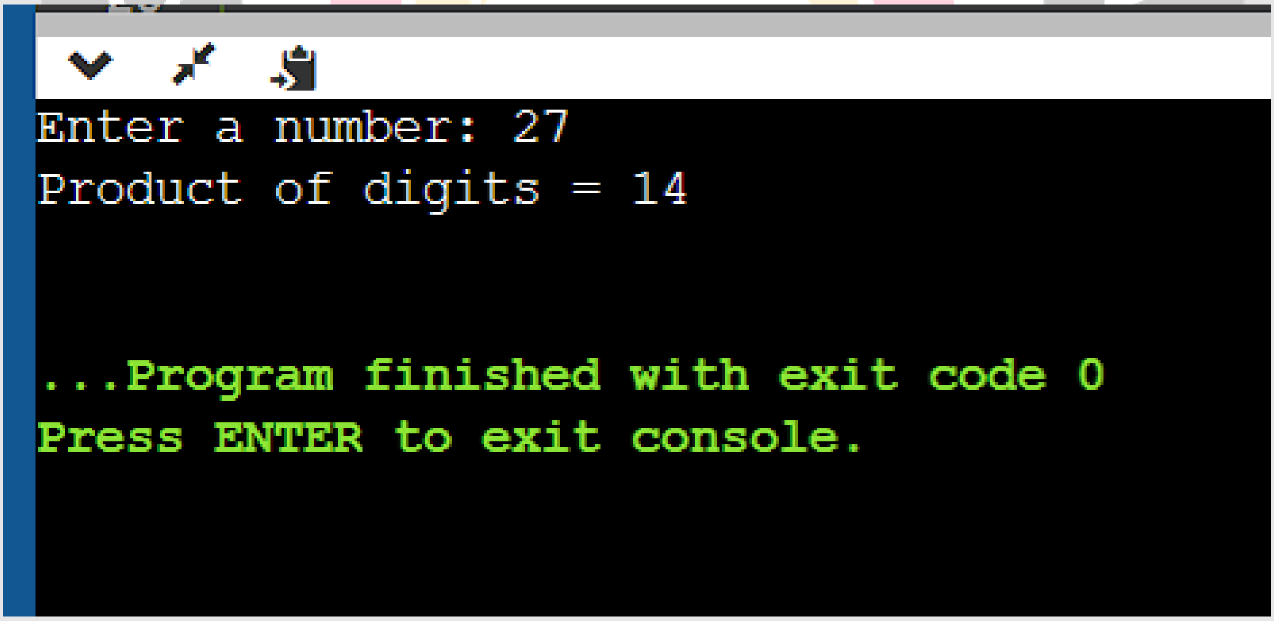
```
#include <stdio.h>

int main() {
    int num, sum = 0, digit;
    printf("Enter an integer: ");
    scanf("%d", &num);
    while (num > 0) {
        digit = num % 10;   // Get the last digit
        sum += digit;       // Add it to the sum
        num = num / 10;     // Remove the last digit
    }
    printf("The sum of the digits is: %d\n", sum);
    return 0;
//Vinay Kumar
}
```



```
Enter an integer: 23
The sum of the digits is: 5


...Program finished with exit code 0
Press ENTER to exit console.
```

37. Write a C program to calculate product of digits of a number

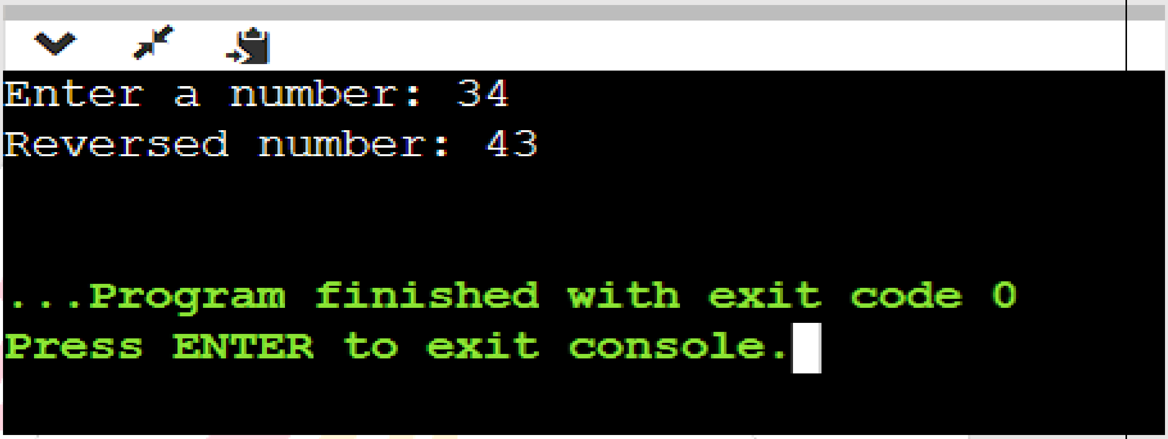
```
#include <stdio.h>

int main() {
    int num, digit, product = 1;

    /* Input number from the user */
    printf("Enter a number: ");
    scanf("%d", &num);

    while (num != 0) {
        digit = num % 10;   // Get the last digit
        product *= digit;   // Multiply it with the product
        num = num / 10;     // Remove the last digit
    }

    printf("Product of digits = %d\n", product);

    return 0;
}
//Vinay Kumar
```



```
Enter a number: 27
Product of digits = 14


...Program finished with exit code 0
Press ENTER to exit console.
```

38. Write a C program to enter a number and print its reverse.

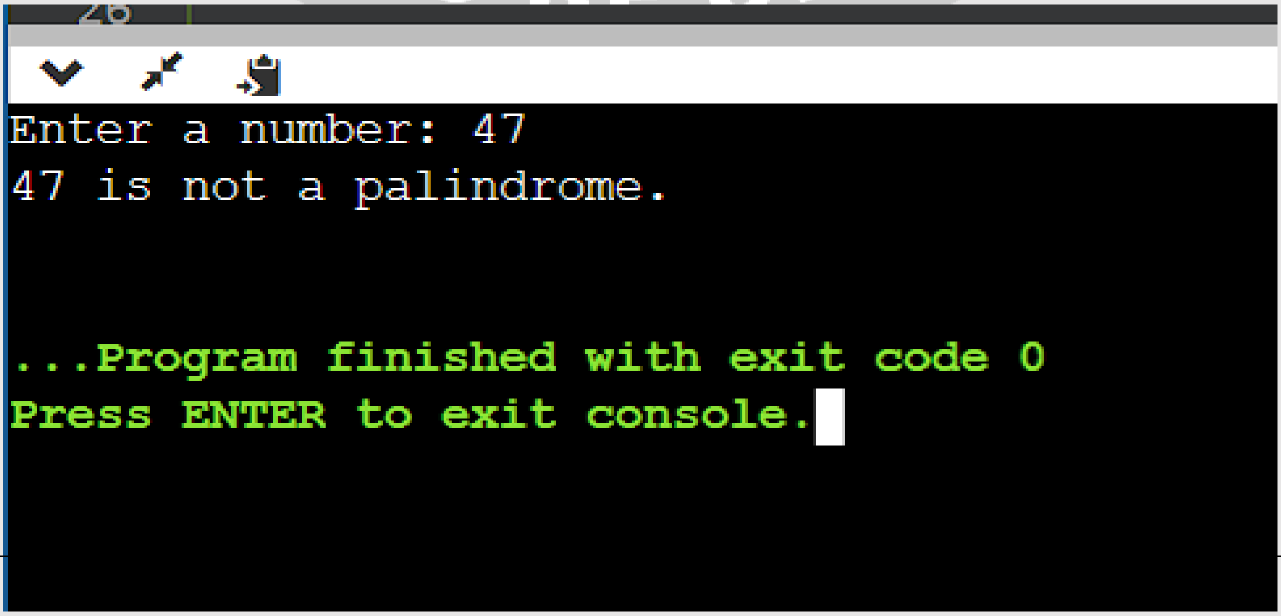
```
#include <stdio.h>


int main() {
    int num, reversed = 0;


    printf("Enter a number: ");
    scanf("%d", &num);


    while (num != 0) {
        int digit = num % 10;
        reversed = reversed * 10 + digit;
        num /= 10;
    }

    printf("Reversed number: %d\n", reversed);
    return 0;

}
//Vinay
Kumar
```



```
Enter a number: 34
Reversed number: 43


...Program finished with exit code 0
Press ENTER to exit console.
```

39. Write a C program to check whether a number is palindrome or not.

```
#include <stdio.h>
int main() {
    int num, originalNum, reversed = 0;

    printf("Enter a number: ");
    scanf("%d", &num);

    originalNum = num;  // Store the original number

    // Reverse the number
    while (num != 0) {
        int digit = num % 10;
        reversed = reversed * 10 + digit;
        num /= 10;
    }

    // Check if the reversed number is the same as the original number
    if (originalNum == reversed) {
        printf("%d is a palindrome.\n", originalNum);
    } else {
        printf("%d is not a palindrome.\n", originalNum);
    }

    return 0;
}
//Vinay Kumar
```

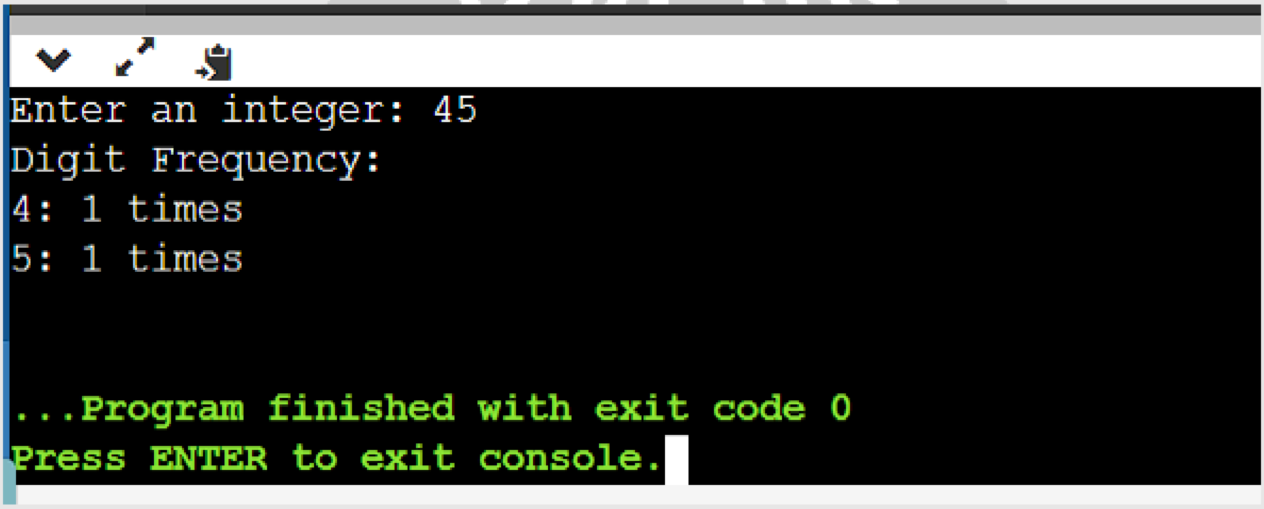


```
Enter a number: 47
47 is not a palindrome.


...Program finished with exit code 0
Press ENTER to exit console.
```

40. Write a C program to find frequency of each digit in a given integer.

```
#include <stdio.h>

int main() {
   int num, digit, count;
   int frequency[10] = {0};  // Initialize an array to store the frequency of each digit

   printf("Enter an integer: ");
   scanf("%d", &num);

   while (num != 0) {
       digit = num % 10;  // Get the last digit
```

```
        frequency[digit]++;  // Increment the count for
that digit
        num /= 10;  // Remove the last digit
    }

    printf("Digit Frequency:\n");
    for (digit = 0; digit <= 9; digit++) {
        if (frequency[digit] > 0) {
            printf("%d: %d times\n", digit,
frequency[digit]);
        }
    }

    return 0;
}

//Vinay Kumar
```

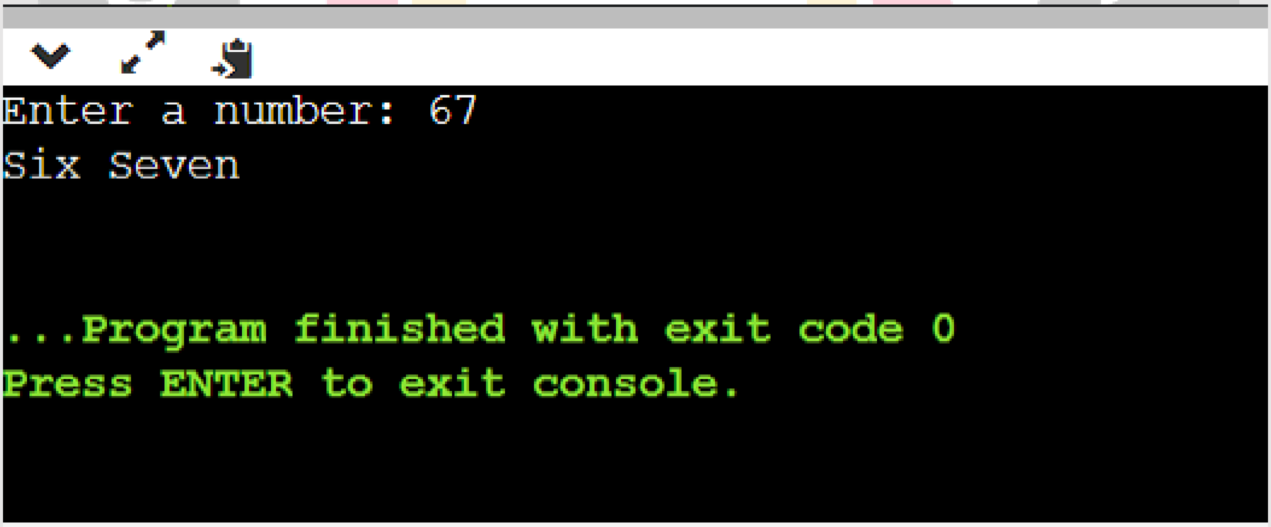


```
Enter an integer: 45
Digit Frequency:
4: 1 times
5: 1 times


...Program finished with exit code 0
Press ENTER to exit console.
```

41. Write a C program to enter a number and print it in words.

```
#include <stdio.h>

// Function to print a number in words for single-digit numbers

void printDigitInWords(int digit) {

    switch (digit) {

        case 0:

            printf("Zero ");

            break;

        case 1:

            printf("One ");
            break;
        case 2:
            printf("Two ");
            break;
        case 3:
            printf("Three ");
            break;
        case 4:
            printf("Four ");
            break;
        case 5:
            printf("Five ");
            break;
        case 6:
            printf("Six ");
            break;
        case 7:
            printf("Seven ");
            break;
        case 8:
            printf("Eight ");
            break;
        case 9:
```

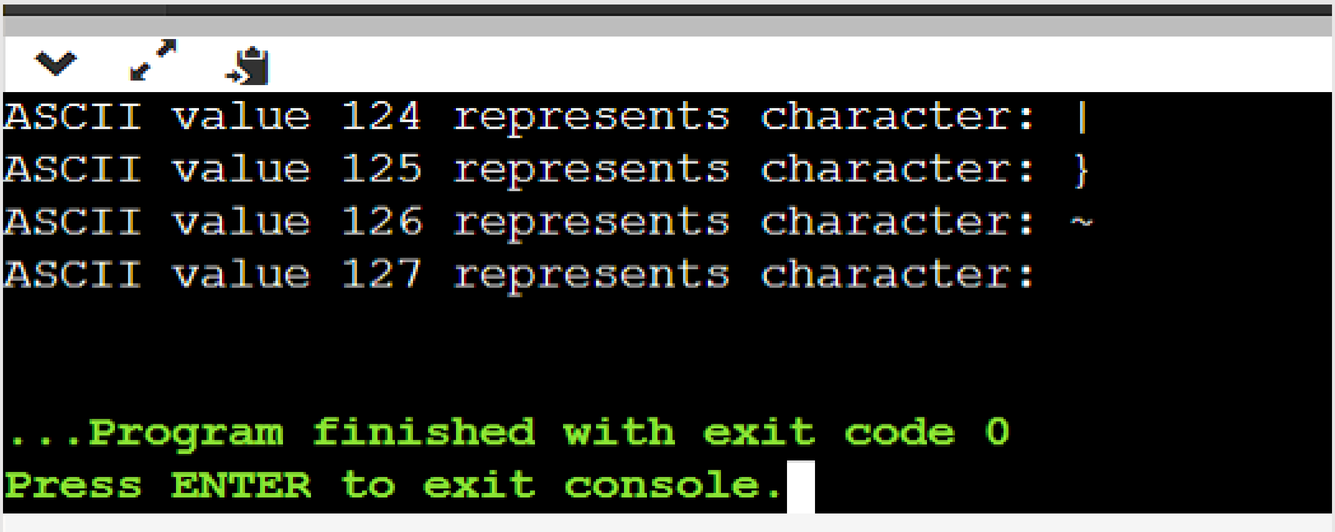
```
            printf("Nine ");
            break;
    }
}

int main() {
    int num, digit, reversed = 0;

    printf("Enter a number: ");
    scanf("%d", &num);

    if (num == 0) {
        printf("Zero\n");
        return 0;
    }

    // Reverse the number to print it correctly
    while (num != 0) {
        digit = num % 10;
        reversed = reversed * 10 + digit;
        num /= 10;
    }
    // Print the number in words
    while (reversed != 0) {
        digit = reversed % 10;
        printDigitInWords(digit);
        reversed /= 10;
    }

    printf("\n");
    return 0;
}//Vinay Kumar
```



```
Enter a number: 67
Six Seven


...Program finished with exit code 0
Press ENTER to exit console.
```

42. Write a C program to print all ASCII character with their values.

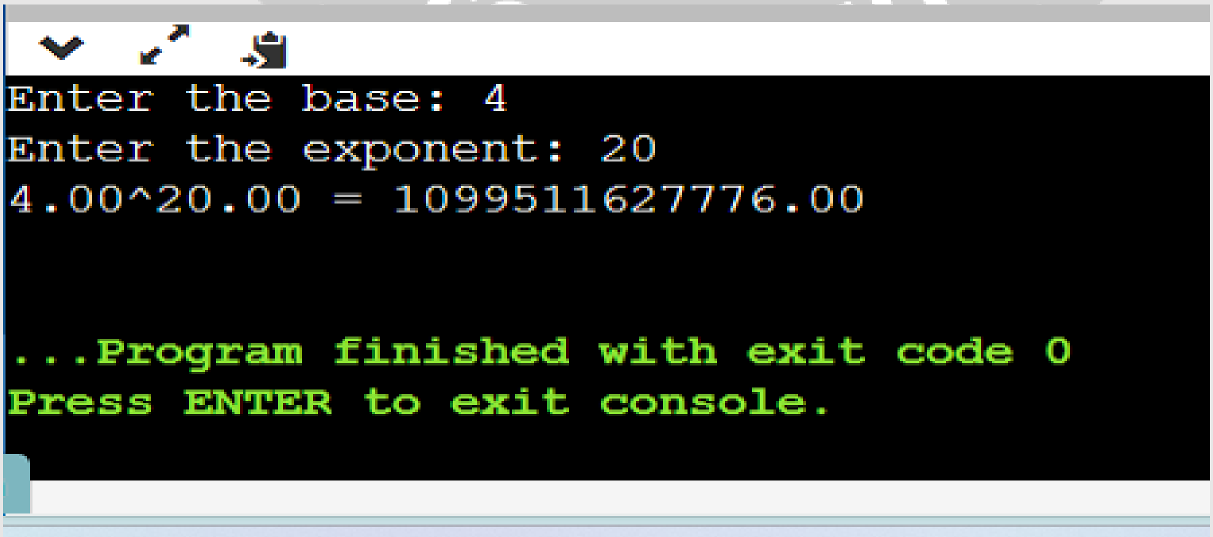
```
#include <stdio.h>


int main() {
    int i;
    for (i = 0; i < 128; i++) {
        printf("ASCII value %d represents character: %c\n", i, i);
    }
    return 0;
}
//Vinay Kumar
```



```
ASCII value 124 represents character: |
ASCII value 125 represents character: }
ASCII value 126 represents character: ~
ASCII value 127 represents character:


...Program finished with exit code 0
Press ENTER to exit console.
```

43. Write a C program to find power of a number using for loop.

```
#include <stdio.h>

int main() {

    double base, exponent, result = 1;

    // Input the base and exponent
    printf("Enter the base: ");
    scanf("%lf", &base);
    printf("Enter the exponent: ");
    scanf("%lf", &exponent);

    // Calculate the power using a for loop
    for (int i = 1; i <= exponent; i++) {
        result *= base;
    }

    // Print the result
    printf("%.2lf^%.2lf = %.2lf\n", base, exponent,
result);

    return 0;
}
//Vinay Kumar
```



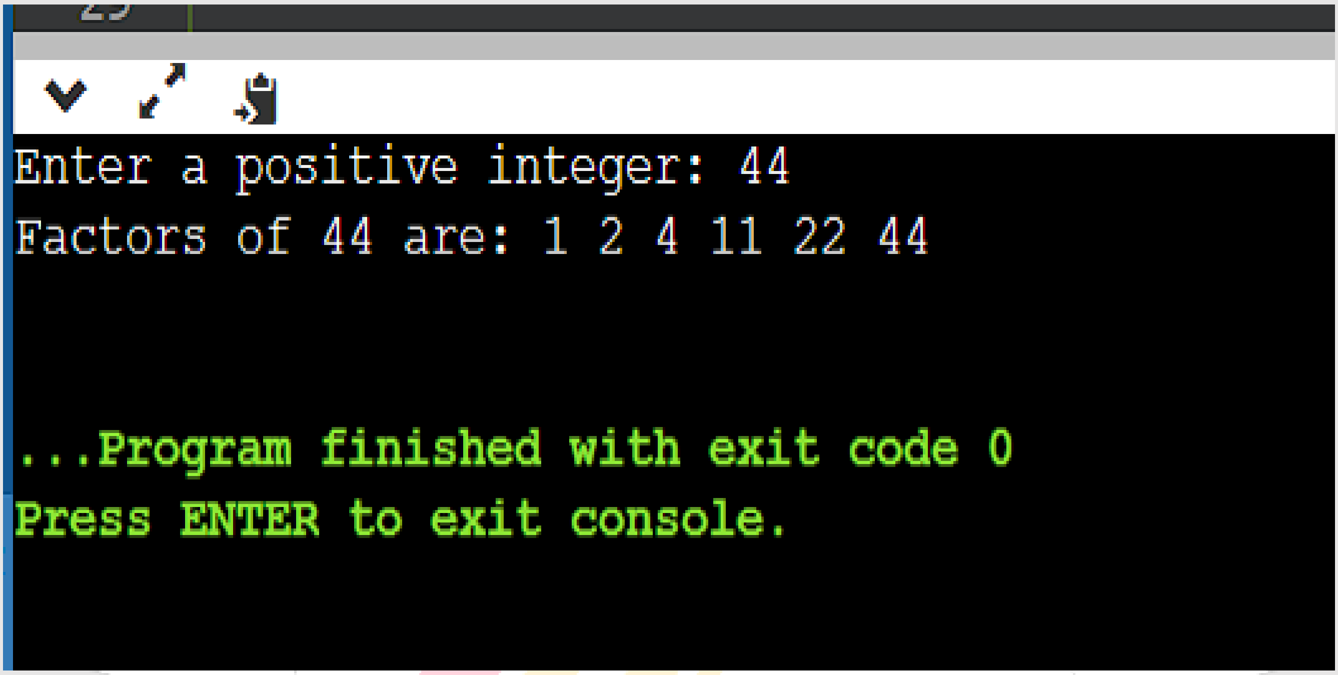
```
Enter the base: 4
Enter the exponent: 20
4.00^20.00 = 1099511627776.00


...Program finished with exit code 0
Press ENTER to exit console.
```

44. Write a C program to find all factors of a number.

```
#include <stdio.h>


int main() {
    int number;


    // Input the number
    printf("Enter a positive integer: ");
    scanf("%d", &number);
```

```
  if (number <= 0) {
    printf("Please enter a positive integer.\n");
    return 1; // Exit with an error code
  }
  printf("Factors of %d are: ", number);

  // Use a for loop to find and print factors
  for (int i = 1; i <= number; i++) {
    if (number % i == 0) {
      printf("%d ", i);
    }
  }

  printf("\n");

  return 0;
//Vinay Kumar
```

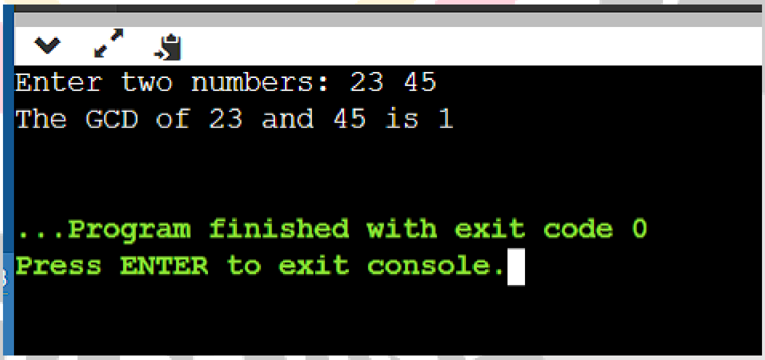
```
}
```



```
Enter a positive integer: 44
Factors of 44 are: 1 2 4 11 22 44


...Program finished with exit code 0
Press ENTER to exit console.
```

46. Write a C program to find HCF (GCD) of two numbers.

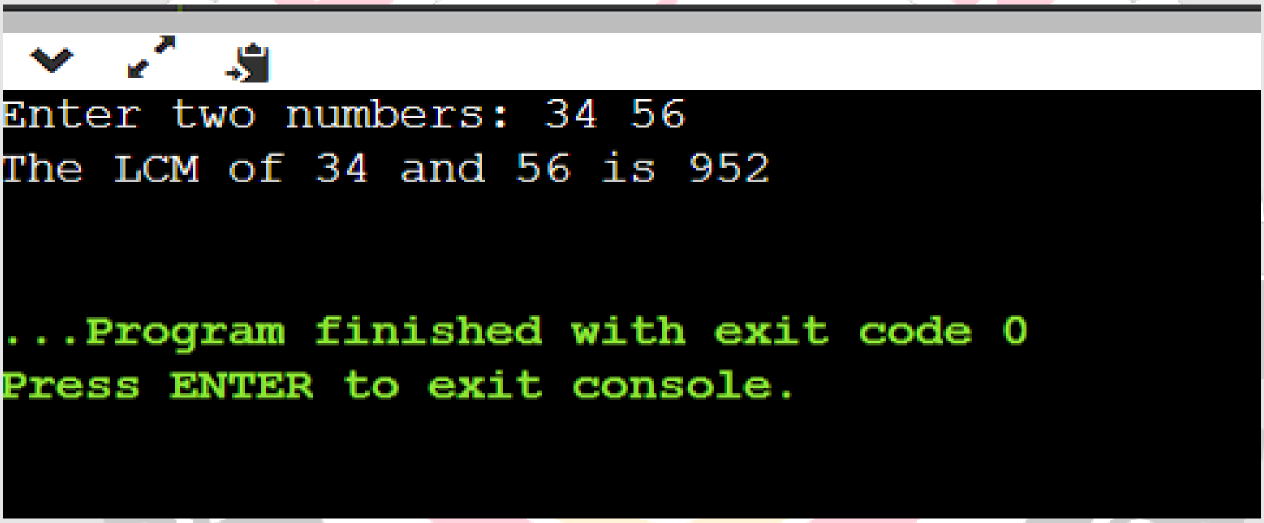
```
#include <stdio.h>
// Function to find the HCF/GCD of two numbers
int findGCD(int a, int b) {
    if (b == 0) {
        return a;
    } else {
        return findGCD(b, a % b);
    }
}
int main() {
    int num1, num2;
    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);

    int gcd = findGCD(num1, num2);

    printf("The GCD of %d and %d is %d\n", num1,
num2, gcd);

    return 0;
}
//Vinay Kumar
```



```
Enter two numbers: 23 45
The GCD of 23 and 45 is 1


...Program finished with exit code 0
Press ENTER to exit console.
```

47. Write a C program to find LCM of two numbers.

```
#include <stdio.h>
// Function to find the GCD (HCF) of two numbers
int findGCD(int a, int b) {
    if (b == 0) {
        return a;
    } else {
        return findGCD(b, a % b);
    }
}

// Function to find the LCM of two numbers
int findLCM(int a, int b) {
    int gcd = findGCD(a, b);
    int lcm = (a * b) / gcd;
    return lcm;
}

int main() {
    int num1, num2;

    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);

    int lcm = findLCM(num1, num2);
    printf("The LCM of %d and %d is %d\n", num1, num2, lcm);


    return 0;
}
//Vinay Kumar
```

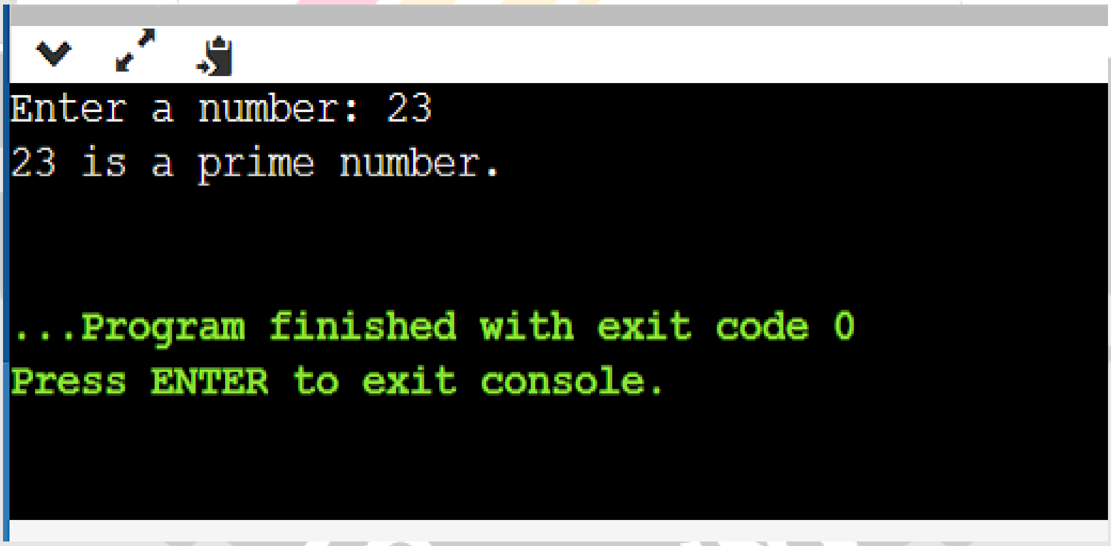


```
Enter two numbers: 34 56
The LCM of 34 and 56 is 952


...Program finished with exit code 0
Press ENTER to exit console.
```

48. Write a C program to check whether a number is Prime number or not.

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

```
#include <stdbool.h>

// Function to check if a number is prime
bool isPrime(int n) {
    if (n <= 1) {
        return false; // 0 and 1 are not prime numbers
    }
    for (int i = 2; i * i <= n; i++) {
        if (n % i == 0) {
            return false; // n is divisible by i, so it's not prime
        }
    }
    return true; // If no divisors are found, it's a prime number
}

int main() {
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);
    if (isPrime(num)) {
        printf("%d is a prime number.\n", num);
    } else {
        printf("%d is not a prime number.\n", num);
    }

    return 0;
}
//Vinay Kumar
```

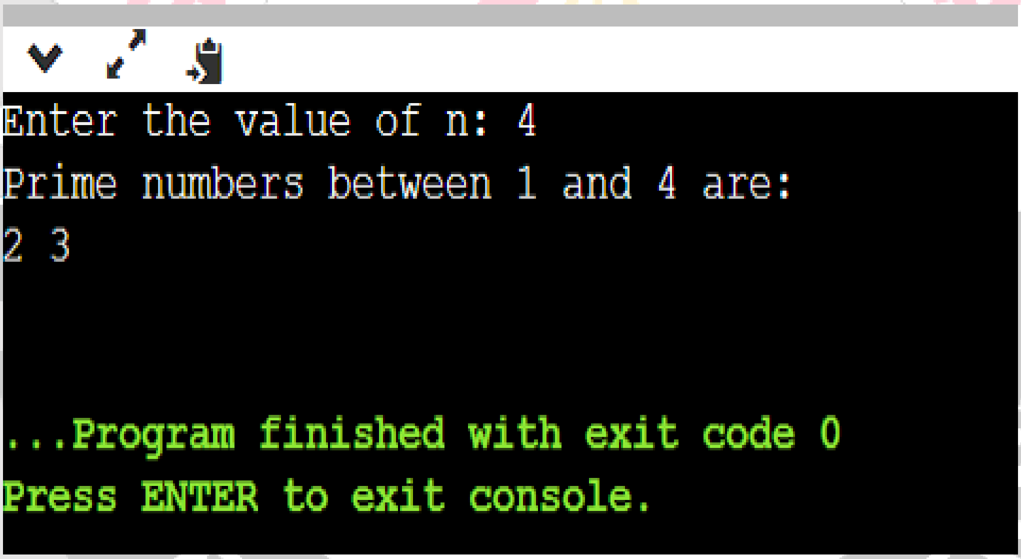


```
Enter a number: 23
23 is a prime number.


...Program finished with exit code 0
Press ENTER to exit console.
```

49. Write a C program to print all Prime numbers between 1 to n.

```
#include <stdio.h>
int isPrime(int num) {
    if (num <= 1) return 0;
    if (num <= 3) return 1;
    if (num % 2 == 0 || num % 3 == 0) return 0;

    for (int i = 5; i * i <= num; i += 6) {
        if (num % i == 0 || num % (i + 2) == 0) {
            return 0;
        }
    }

    return 1;
}

int main() {
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);

    printf("Prime numbers between 1 and %d are:\n",
n);

    for (int i = 2; i <= n; i++) {
        if (isPrime(i)) {
            printf("%d ", i);
        }
    }

    printf("\n");
    return 0;
}
```

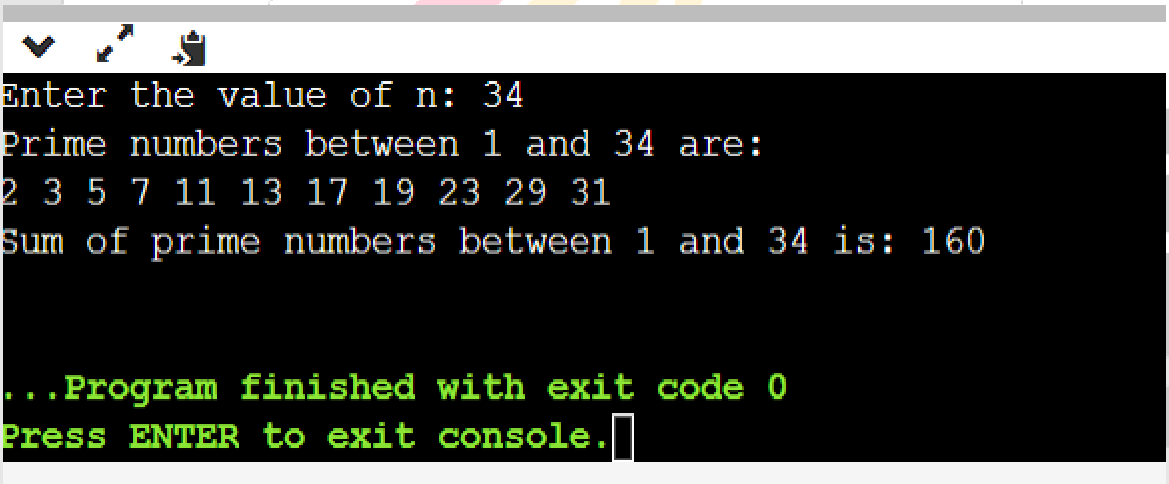
//Vinay Kumar



```
Enter the value of n: 4
Prime numbers between 1 and 4 are:
2 3


...Program finished with exit code 0
Press ENTER to exit console.
```

50. Write a C program to find sum of all prime numbers between 1 to n.

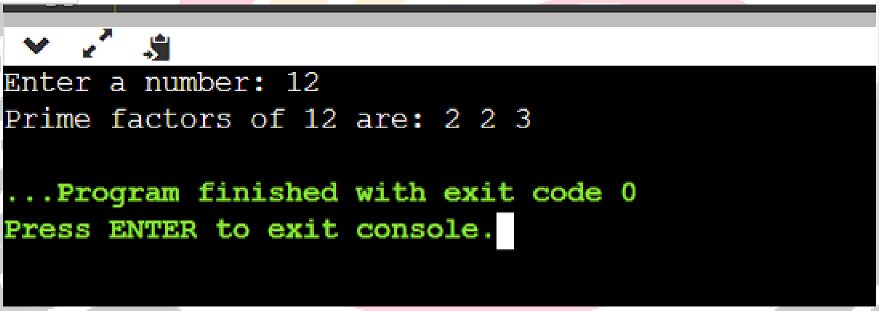
```
#include <stdio.h>

int isPrime(int num) {
    if (num <= 1) return 0;
    if (num <= 3) return 1;
    if (num % 2 == 0 || num % 3 == 0) return 0;

    for (int i = 5; i * i <= num; i += 6) {
        if (num % i == 0 || num % (i + 2) == 0) {
            return 0;
        }
    }

    return 1;
}
int main() {
    int n;
    printf("Enter the value of n: ");
    scanf("%d", &n);

    int sum = 0;
    printf("Prime numbers between 1 and %d are:\n",
n);

    for (int i = 2; i <= n; i++) {
        if (isPrime(i)) {
            printf("%d ", i);
            sum += i;
        }
    }

    printf("\nSum of prime numbers between 1 and %d is: %d\n", n, sum);
    return 0;
}
//Vinay Kumar
```



```
Enter the value of n: 34
Prime numbers between 1 and 34 are:
2 3 5 7 11 13 17 19 23 29 31
Sum of prime numbers between 1 and 34 is: 160


...Program finished with exit code 0
Press ENTER to exit console.
```

51. Write a C program to find all prime factors of a number.

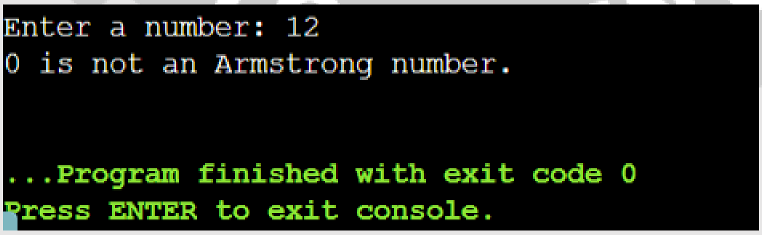
```
#include <stdio.h>
// Function to find and print all prime factors of a number
void primeFactors(int n) {
    // Print the number of 2s that divide n
    while (n % 2 == 0) {
        printf("2 ");
        n = n / 2;
    }

    // n must be odd at this point, so a skip of 2 ( i = i + 2) can be used
    for (int i = 3; i * i <= n; i = i + 2) {
        // While i divides n, print i and divide n
        while (n % i == 0) {
            printf("%d ", i);
            n = n / i;
        }
    }

    // If n is a prime greater than 2
    if (n > 2) {
        printf("%d ", n);
    }
}

int main() {
    int n;
    printf("Enter a number: ");
    scanf("%d", &n);

    printf("Prime factors of %d are: ", n);
    primeFactors(n);

    return 0;
}
//Vinay Kumar
```



```
Enter a number: 12
Prime factors of 12 are: 2 2 3

...Program finished with exit code 0
Press ENTER to exit console.
```

52. Write a C program to check whether a number is Armstrong number or not.

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

int isArmstrong(int num) {
    int originalNum, remainder, n = 0, result = 0;

    originalNum = num;

    while (originalNum != 0) {
        originalNum /= 10;
        ++n;
    }

    originalNum = num;

    while (originalNum != 0) {
        remainder = originalNum % 10;
        result += pow(remainder, n);
        originalNum /= 10;
    }

    if (result == num)
        return 1; // It's an Armstrong number
    else
        return 0; // It's not an Armstrong number
}

int main() {
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);

    if (isArmstrong(num))
        printf("%d is an Armstrong number.\n");
    else
        printf("%d is not an Armstrong number.\n");

    return 0;
}
```



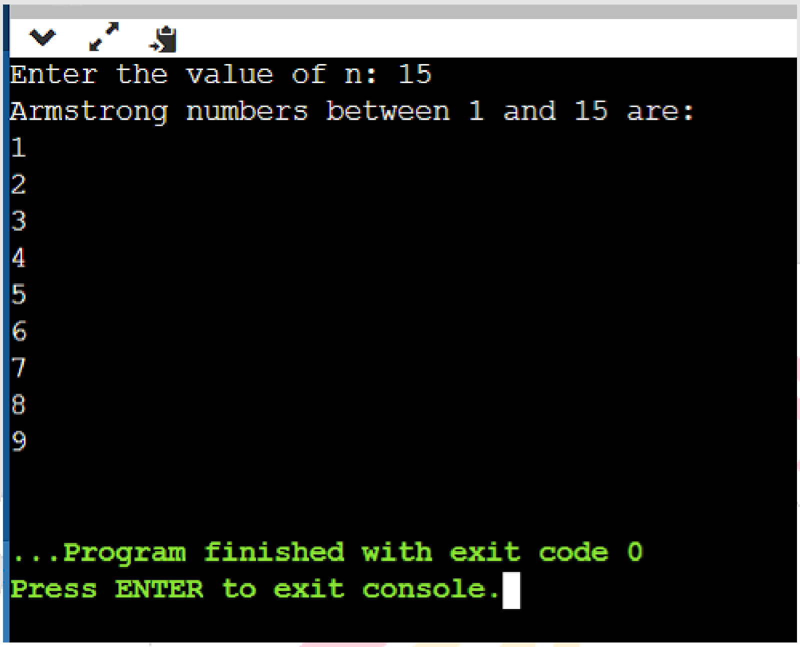
```
Enter a number: 12
0 is not an Armstrong number.


...Program finished with exit code 0
Press ENTER to exit console.
```

//Vinay Kumar

53. Write a C program to print all Armstrong numbers between 1 to n.

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

int isArmstrong(int num) {
    int originalNum, remainder, result = 0, n = 0;

    originalNum = num;

    while (originalNum != 0) {
        originalNum /= 10;
        ++n;
    }

    originalNum = num;

    while (originalNum != 0) {
        remainder = originalNum % 10;
        result += pow(remainder, n);
        originalNum /= 10;
    }
    if (result == num)
        return 1;
    else
        return 0;
}

int main() {
    int n, i;

    printf("Enter the value of n: ");
    scanf("%d", &n);

    printf("Armstrong numbers between 1 and %d
are:\n", n);

    for (i = 1; i <= n; i++) {
        if (isArmstrong(i))
            printf("%d\n", i);
    }

    return 0;
//Vinay Kumar
```

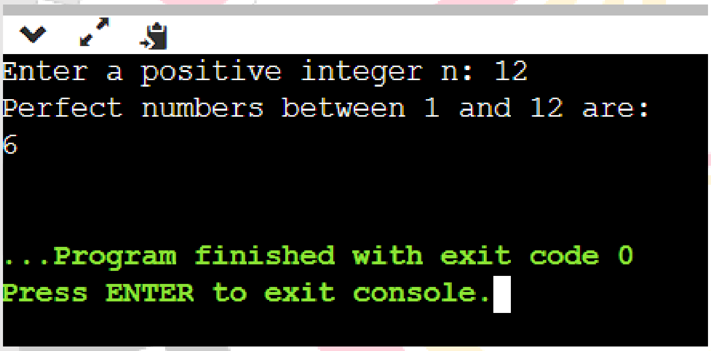
```
}
```



```
Enter the value of n: 15
Armstrong numbers between 1 and 15 are:
1
2
3
4
5
6
7
8
9


...Program finished with exit code 0
Press ENTER to exit console.
```

55. Write a C program to print all Perfect numbers between 1 to n.

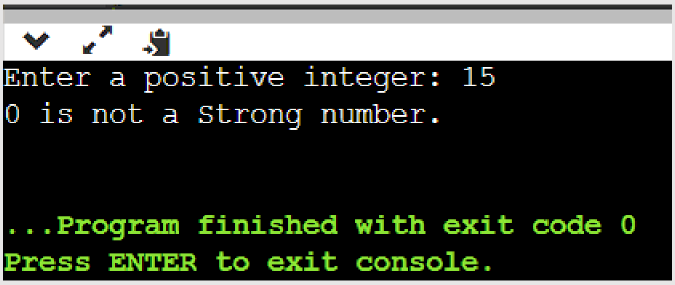
```
#include <stdio.h>
int isPerfect(int num) {
    int sum = 0;
    for (int i = 1; i <= num / 2; i++) {
        if (num % i == 0) {
            sum += i;
        }
    }
    return (sum == num);
}

int main() {
    int n;
    printf("Enter a positive integer n: ");
    scanf("%d", &n);

    printf("Perfect numbers between 1 and %d are: \n", n);

    for (int i = 1; i <= n; i++) {
        if (isPerfect(i)) {
            printf("%d\n", i);
        }
    }

    return 0;
}
//Vinay Kumar
```



```
Enter a positive integer n: 12
Perfect numbers between 1 and 12 are:
6


...Program finished with exit code 0
Press ENTER to exit console.
```

56. Write a C program to check whether a number is Strong number or not.

```
#include <stdio.h>
// Function to calculate the factorial of a number
int factorial(int num) {
    int fact = 1;
    for (int i = 1; i <= num; i++) {
        fact *= i;
    }
    return fact;
}

// Function to check if a number is a strong number
int isStrongNumber(int num) {
    int originalNum = num;
    int sum = 0;
    while (num > 0) {
        int digit = num % 10;
        sum += factorial(digit);
        num /= 10;
    }
    return (sum == originalNum);
}

int main() {
    int n;
    printf("Enter a positive integer: ");
    scanf("%d", &n);

    if (isStrongNumber(n)) {
        printf("%d is a Strong number.\n");
    } else {
        printf("%d is not a Strong number.\n");
    }

    return 0;
}
//Vinay Kumar
```

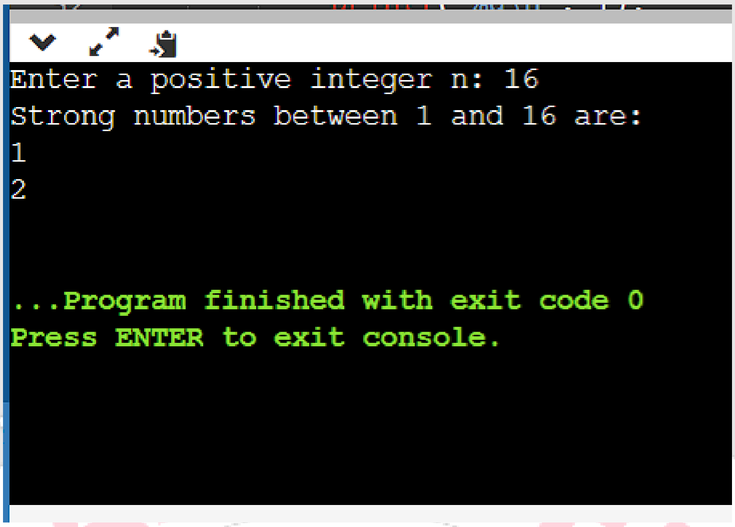


```
Enter a positive integer: 15
0 is not a Strong number.


...Program finished with exit code 0
Press ENTER to exit console.
```

57. Write a C program to print all Strong numbers between 1 to n.

```
#include <stdio.h>
// Function to calculate the factorial of a number
int factorial(int num) {
    int fact = 1;
    for (int i = 1; i <= num; i++) {
        fact *= i;
    }
    return fact;
}

// Function to check if a number is a strong number
int isStrongNumber(int num) {
    int originalNum = num;
    int sum = 0;
    while (num > 0) {
        int digit = num % 10;
        sum += factorial(digit);
        num /= 10;
    }
    return (sum == originalNum);
}

int main() {
    int n;
    printf("Enter a positive integer n: ");
    scanf("%d", &n);

    printf("Strong numbers between 1 and %d are: \n", n);

    for (int i = 1; i <= n; i++) {
        if (isStrongNumber(i)) {
            printf("%d\n", i);
        }
    }

    return 0;
}
```

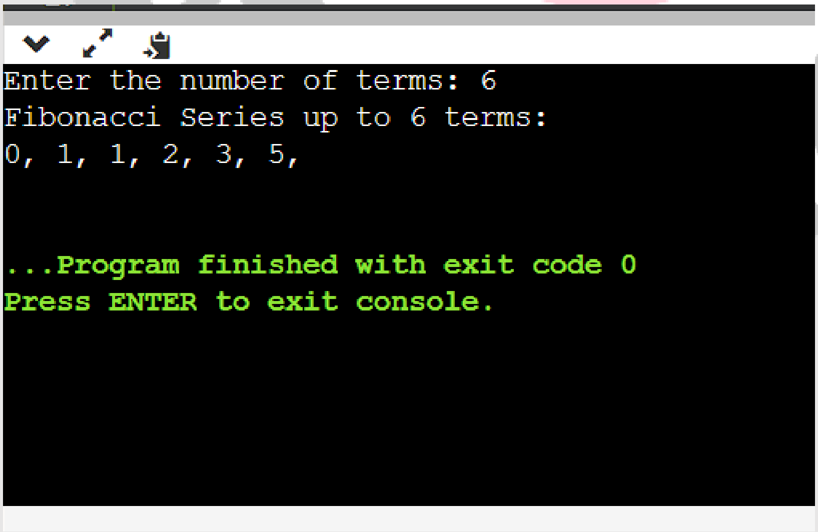
//Vinay Kumar



```
Enter a positive integer n: 16
Strong numbers between 1 and 16 are:
1
2


...Program finished with exit code 0
Press ENTER to exit console.
```

58. Write a C program to print Fibonacci series up to n terms.

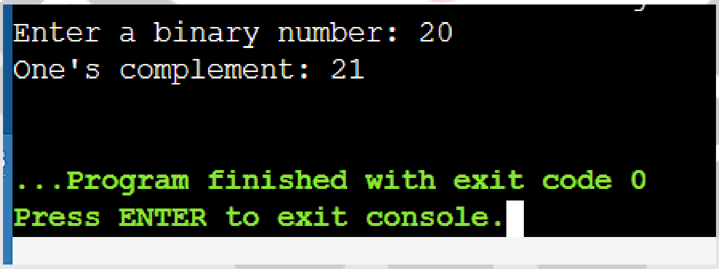
```
#include <stdio.h>
int main() {
  int n, first = 0, second = 1, next;

  printf("Enter the number of terms: ");
  scanf("%d", &n);

  printf("Fibonacci Series up to %d terms: \n", n);

  for (int i = 1; i <= n; i++) {
    if (i == 1) {
      printf("%d, ", first);
    } else if (i == 2) {
            printf("%d, ", second);
        } else {
            next = first + second;
            printf("%d, ", next);
            first = second;
            second = next;
        }
    }

    printf("\n");

    return 0;
}
//Vinay Kumar
```



```
Enter the number of terms: 6
Fibonacci Series up to 6 terms:
0, 1, 1, 2, 3, 5,


...Program finished with exit code 0
Press ENTER to exit console.
```

59. Write a C program to find one's complement of a binary number.

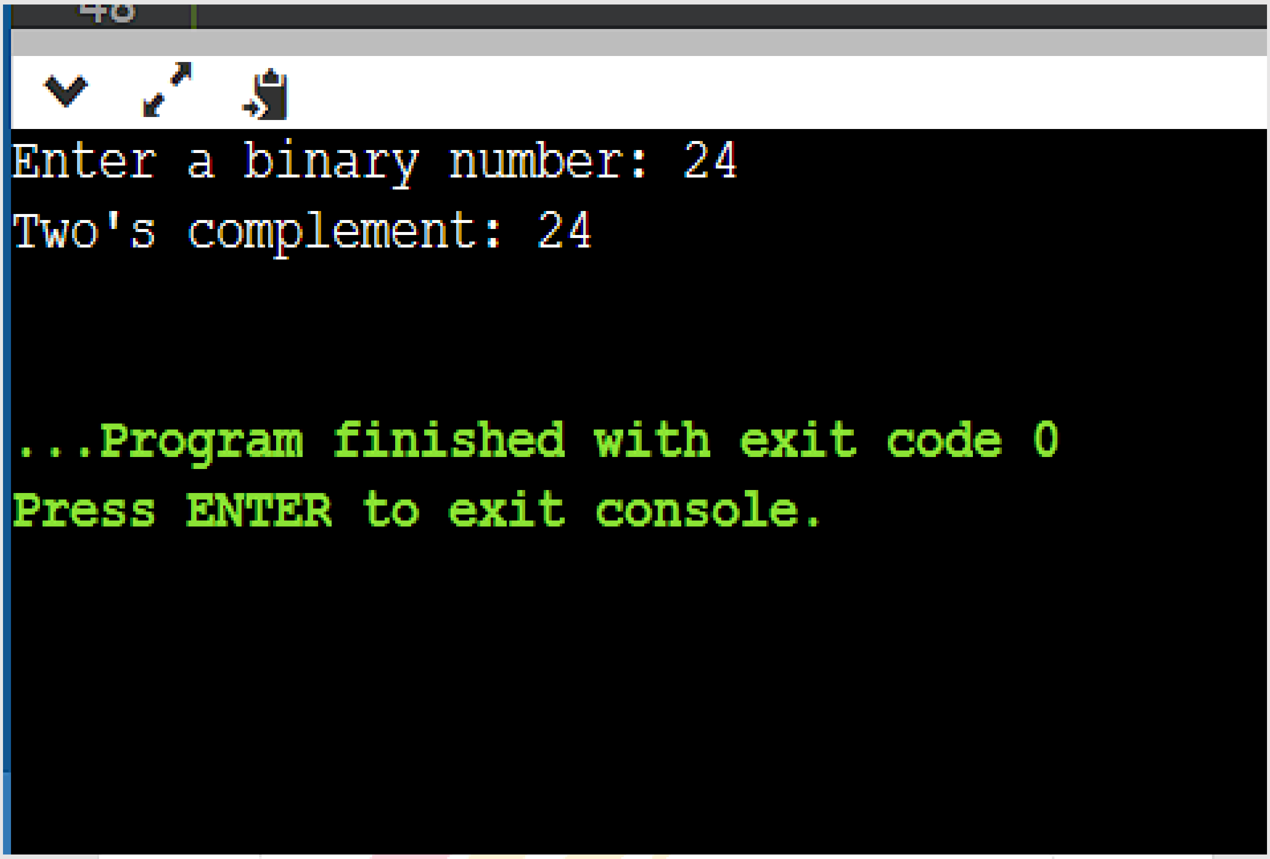
```
#include <stdio.h>
int main() {
    char binary[32]; // Assuming a maximum of 32-bit binary number
    int length, i;

    printf("Enter a binary number: ");
    scanf("%s", binary);

    // Find the length of the binary number
    length = strlen(binary);
    // Perform one's complement
    for (i = 0; i < length; i++) {
        if (binary[i] == '0') {
            binary[i] = '1';
        } else if (binary[i] == '1') {
            binary[i] = '0';
        }
    }

    printf("One's complement: %s\n", binary);

    return 0;
}
//Vinay Kumar
```



```
Enter a binary number: 20
One's complement: 21


...Program finished with exit code 0
Press ENTER to exit console.
```

60. Write a C program to find two's complement of a binary number.

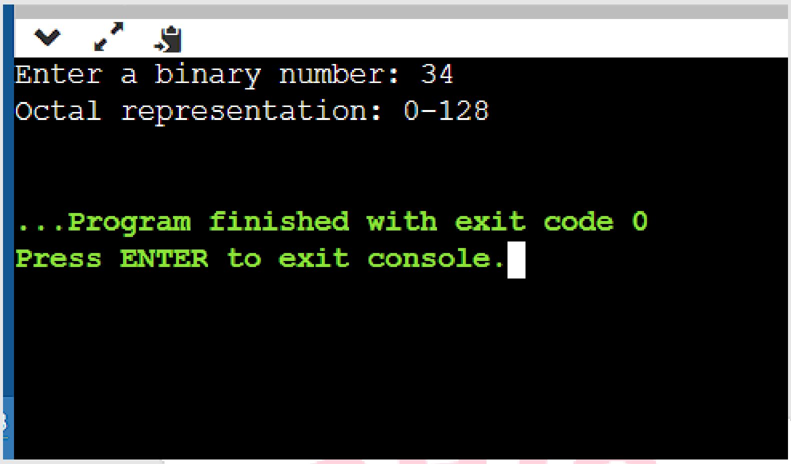
```
#include <stdio.h>

#include <string.h>


// Function to reverse a binary string

void reverseString(char str[]) {

    int length = strlen(str);

    for (int i = 0; i < length / 2; i++) {

        char temp = str[i];

        str[i] = str[length - i - 1];

        str[length - i - 1] = temp;
    }
}

// Function to add 1 to a binary string
void addOne(char binary[]) {
    int length = strlen(binary);
    int carry = 1;

    for (int i = 0; i < length; i++) {
        if (binary[i] == '0' && carry == 1) {
            binary[i] = '1';
            carry = 0;
        } else if (binary[i] == '1' && carry == 1) {
            binary[i] = '0';
        }
    }
}

int main() {
    char binary[32]; // Assuming a maximum of 32-bit
binary number


    printf("Enter a binary number: ");
    scanf("%s", binary);

    // Reverse the binary string
    reverseString(binary);

    // Add 1 to the reversed binary string
    addOne(binary);

    // Reverse the result to get the two's complement
    reverseString(binary);

    printf("Two's complement: %s\n", binary);

    return 0;
}
//Vinay Kumar
```



```
Enter a binary number: 24
Two's complement: 24


...Program finished with exit code 0
Press ENTER to exit console.
```

61. Write a C program to convert Binary to Octal number system.

```
#include <stdio.h>

#include <string.h>


// Function to convert a binary number to an octal number

void binaryToOctal(char binary[]) {

    int length = strlen(binary);


    // Pad the binary number with leading zeros if needed to make the length a multiple of 3

    int padding = (3 - (length % 3)) % 3;

    for (int i = 0; i < padding; i++) {

        printf("0");

    }
    // Iterate through the binary number in groups of 3 and convert to octal
    for (int i = padding; i < length; i += 3) {
        int octalDigit = (binary[i] - '0') * 4 + (binary[i + 1] - '0') * 2 + (binary[i + 2] - '0');
        printf("%d", octalDigit);
    }
    printf("\n");
}

int main() {
    char binary[32]; // Assuming a maximum of 32-bit binary number

    printf("Enter a binary number: ");
    scanf("%s", binary);

    printf("Octal representation: ");
    binaryToOctal(binary);

    return 0;
}
//Vinay Kumar
```

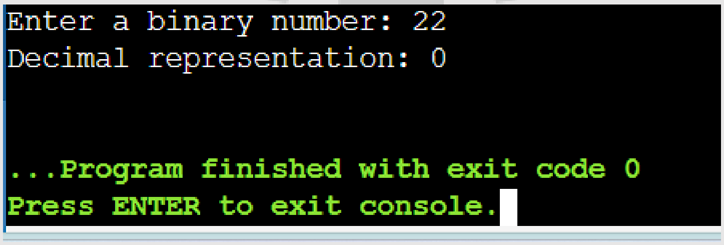


```
Enter a binary number: 34
Octal representation: 0-128


...Program finished with exit code 0
Press ENTER to exit console.
```

62. Write a C program to convert Binary to Decimal number system.#include <stdio.h>

```
#include <string.h>
// Function to convert a binary number to decimal
int binaryToDecimal(char binary[]) {
  int length = strlen(binary);
  int decimal = 0;
  int base = 1;

  for (int i = length - 1; i >= 0; i--) {
    if (binary[i] == '1') {
      decimal += base;
    }
    base *= 2;
  }
```

```
    return decimal;
}

int main() {
    char binary[32]; // Assuming a maximum of 32-bit binary number

    printf("Enter a binary number: ");
    scanf("%s", binary);

    int decimal = binaryToDecimal(binary);

    printf("Decimal representation: %d\n", decimal);

    return 0;
}
//Vinay Kumar
```

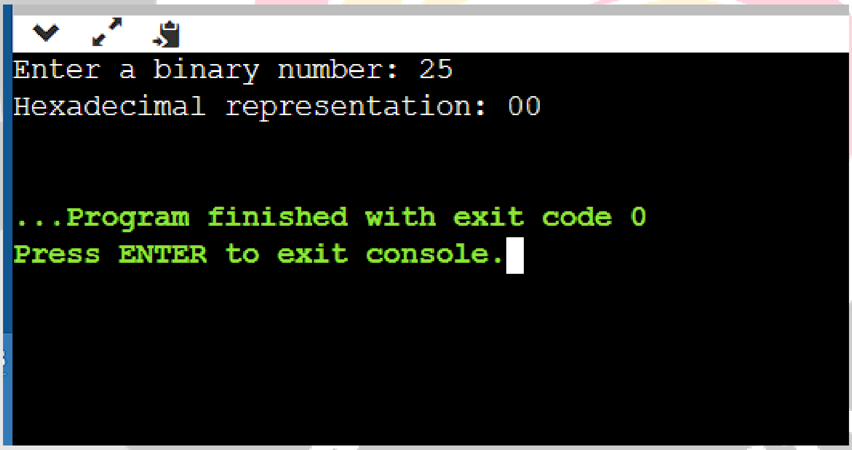


```
Enter a binary number: 22
Decimal representation: 0


...Program finished with exit code 0
Press ENTER to exit console.
```

63. Write a C program to convert Binary to Hexadecimal number system.

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

// Function to convert a binary number to
hexadecimal
void binaryToHexadecimal(char binary[]) {
  int length = strlen(binary);

  // Pad the binary number with leading zeros if
needed to make the length a multiple of 4
  int padding = (4 - (length % 4)) % 4;
  for (int i = 0; i < padding; i++) {
    printf("0");
  }
```

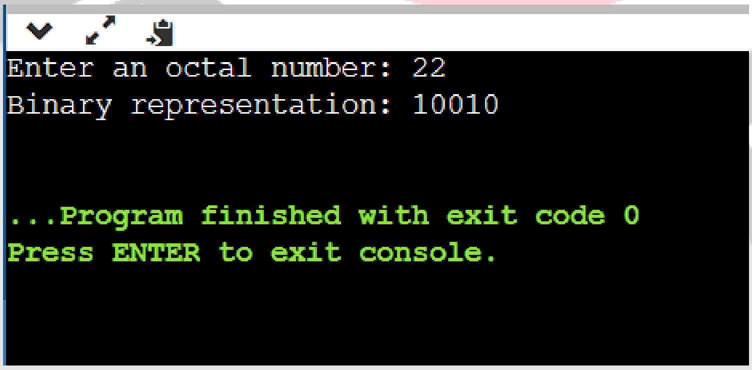
```
    // Iterate through the binary number in groups of
4 and convert to hexadecimal
    for (int i = padding; i < length; i += 4) {
        int hexDigit = 0;
        for (int j = 0; j < 4; j++) {
            hexDigit = (hexDigit << 1) | (binary[i + j] - '0');
        }

        if (hexDigit < 10) {
            printf("%d", hexDigit);
        } else {
            printf("%c", 'A' + hexDigit - 10);
        }
    }
    printf("\n");
}

int main() {
    char binary[32]; // Assuming a maximum of 32-bit
binary number

    printf("Enter a binary number: ");
    scanf("%s", binary);

    printf("Hexadecimal representation: ");
    binaryToHexadecimal(binary);

    return 0;


}
//Vinay Kumar
```



```
Enter a binary number: 25
Hexadecimal representation: 00


...Program finished with exit code 0
Press ENTER to exit console.
```

64. Write a C program to convert Octal to Binary number system.

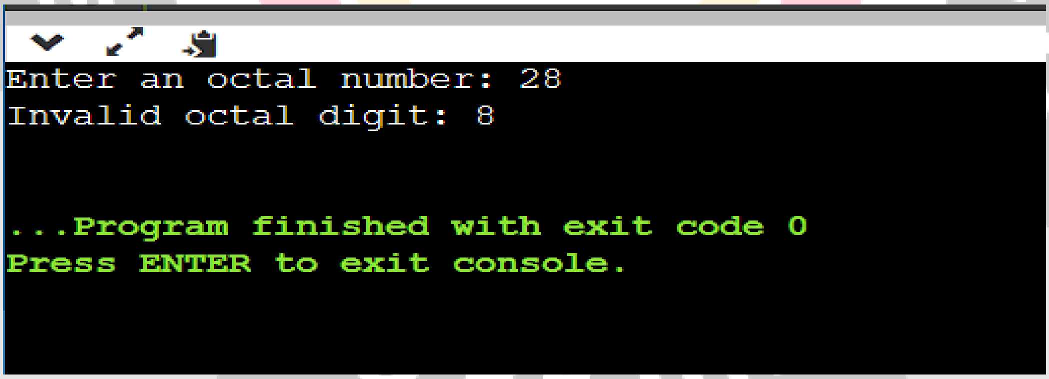
```
#include <stdio.h>
#include <string.h>

// Function to convert an octal digit to a binary string
char *octalToBinary(char octalDigit) {
    switch (octalDigit) {
        case '0': return "000";
        case '1': return "001";
        case '2': return "010";
        case '3': return "011";
        case '4': return "100";
        case '5': return "101";
        case '6': return "110";
        case '7': return "111";
        default: return "Invalid";
    }
}

int main() {
    char octal[32]; // Assuming a maximum of 32-bit octal number
    char binary[128]; // To store the binary equivalent

    printf("Enter an octal number: ");
    scanf("%s", octal);

    int length = strlen(octal);
    int binaryIndex = 0;

    // Convert each octal digit to binary and concatenate
    for (int i = 0; i < length; i++) {
        char *binaryDigit = octalToBinary(octal[i]);
        // Skip the first digit if it results in leading zeros
        if (i == 0 && binaryDigit[0] == '0') {
            binaryDigit += 1;
        }

        // Copy the binary digits to the binary string
        strcpy(binary + binaryIndex, binaryDigit);
        binaryIndex += strlen(binaryDigit);
    }

    printf("Binary representation: %s\n", binary);

    return 0;
}
//Vinay Kumar
```



```
Enter an octal number: 22
Binary representation: 10010


...Program finished with exit code 0
Press ENTER to exit console.
```

65. Write a C program to convert Octal to Decimal number system.

```
#include <stdio.h>
#include <math.h>
int octalToDecimal(char octal[]) {
    int length = 0;
    while (octal[length] != '\0') {
        length++;
    }

    int decimal = 0;
    int base = 1;

    for (int i = length - 1; i >= 0; i--) {
        if (octal[i] < '0' || octal[i] > '7') {
            printf("Invalid octal digit: %c\n", octal[i]);
            return -1;
        }

        int octalDigit = octal[i] - '0';
        decimal += octalDigit * base;
        base *= 8;
    }

    return decimal;
}

int main() {
    char octal[32]; // Assuming a maximum of 32-bit
octal number

    printf("Enter an octal number: ");
    scanf("%s", octal);
    int decimal = octalToDecimal(octal);


    if (decimal != -1) {
        printf("Decimal representation: %d\n",
decimal);
    }


    return 0;
}
//Vinay Kumar
```

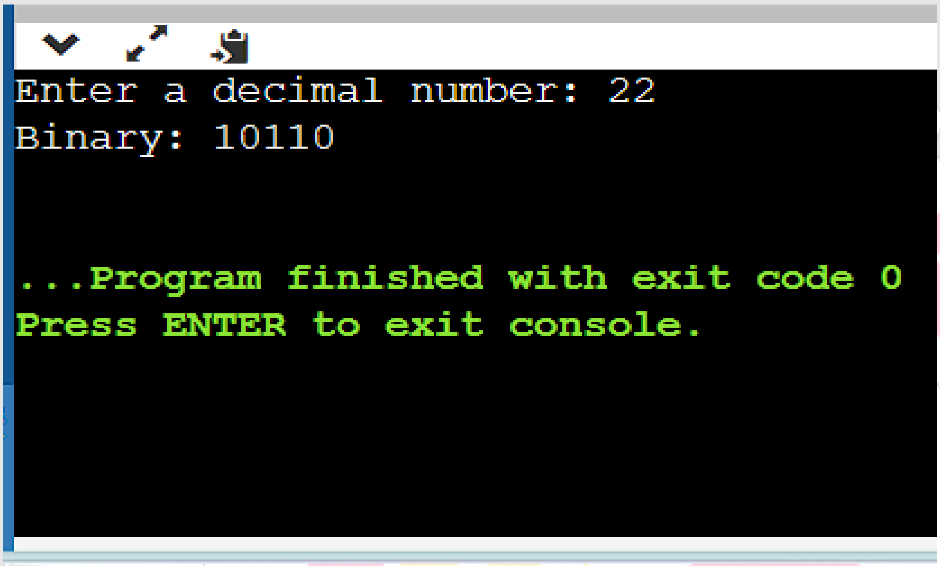


```
Enter an octal number: 28
Invalid octal digit: 8


...Program finished with exit code 0
Press ENTER to exit console.
```

67. Write a C program to convert Decimal to Binary number system.

```
#include <stdio.h>
void decimalToBinary(int decimal) {
  if (decimal == 0) {
    printf("Binary: 0\n");
    return;
  }

  int binary[32];
```

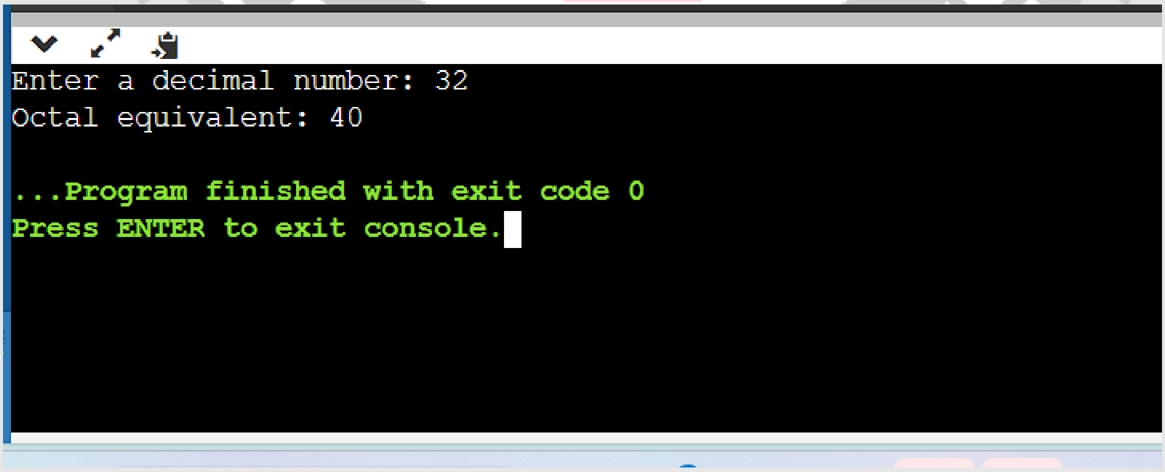
```
    int i = 0;

    while (decimal > 0) {
        binary[i] = decimal % 2;
        decimal /= 2;
        i++;
    }

    printf("Binary: ");
    for (int j = i - 1; j >= 0; j--) {
        printf("%d", binary[j]);
    }
    printf("\n");
}

int main() {
    int decimal;

    printf("Enter a decimal number: ");
    scanf("%d", &decimal);

    decimalToBinary(decimal);
    return 0;
}
//Vinay Kumar
```



```
Enter a decimal number: 22
Binary: 10110


...Program finished with exit code 0
Press ENTER to exit console.
```

68. Write a C program to convert Decimal to Octal number system.

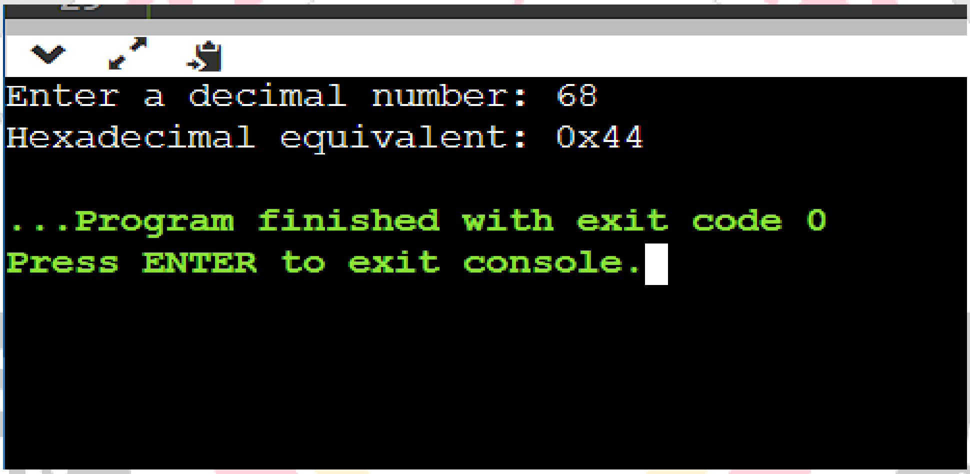
```
#include <stdio.h>
int main() {
    int decimal, octal[100], i = 0;

    printf("Enter a decimal number: ");
    scanf("%d", &decimal);
    while (decimal > 0) {
        octal[i] = decimal % 8;
        decimal /= 8;
        i++;
    }

    printf("Octal equivalent: ");
    for (int j = i - 1; j >= 0; j--) {
        printf("%d", octal[j]);
    }

    return 0;
}
//Vinay Kumar
```



```
Enter a decimal number: 32
Octal equivalent: 40

...Program finished with exit code 0
Press ENTER to exit console.
```

69. Write a C program to convert Decimal to Hexadecimal number system.

```
#include <stdio.h>

int main() {

    int decimal, remainder, i = 0;
    char hexadecimal[100];

    printf("Enter a decimal number: ");
    scanf("%d", &decimal);

    while (decimal > 0) {
        remainder = decimal % 16;
        if (remainder < 10) {
            hexadecimal[i] = remainder + '0';
        } else {
            hexadecimal[i] = remainder - 10 + 'A';
        }
        decimal /= 16;
        i++;
    }

    printf("Hexadecimal equivalent: 0x");
    for (int j = i - 1; j >= 0; j--) {
        printf("%c", hexadecimal[j]);
    }

    return 0;
}

//Vinay Kumar
```

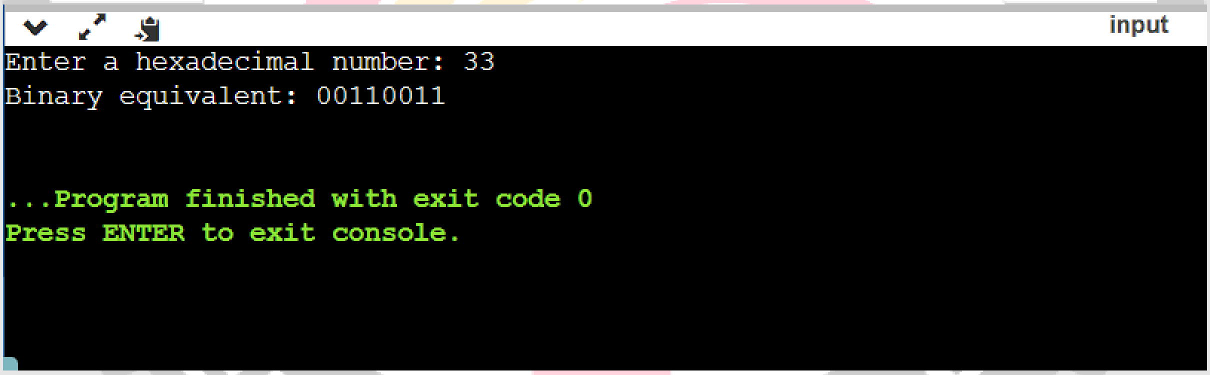


```
Enter a decimal number: 68
Hexadecimal equivalent: 0x44

...Program finished with exit code 0
Press ENTER to exit console.
```

70. Write a C program to convert Hexadecimal to Binary number system.

```
#include <stdio.h>

#include <string.h>


int main() {

    char hexadecimal[100];

    printf("Enter a hexadecimal number: ");
```

```
scanf("%s", hexadecimal);

int length = strlen(hexadecimal);
printf("Binary equivalent: ");

for (int i = 0; i < length; i++) {
    char hexDigit = hexadecimal[i];
    int decimalValue;

    if (hexDigit >= '0' && hexDigit <= '9') {
        decimalValue = hexDigit - '0';
    } else if (hexDigit >= 'A' && hexDigit <= 'F') {
        decimalValue = hexDigit - 'A' + 10;
    } else if (hexDigit >= 'a' && hexDigit <= 'f') {
        decimalValue = hexDigit - 'a' + 10;
    } else {
        printf("Invalid hexadecimal input.\n");
        return 1;
    }

    for (int j = 3; j >= 0; j--) {
        if (decimalValue & (1 << j)) {
            printf("1");
            } else {
                printf("0");
            }
        }
    }

    printf("\n");
    return 0;
}
//Vinay Kumar
```



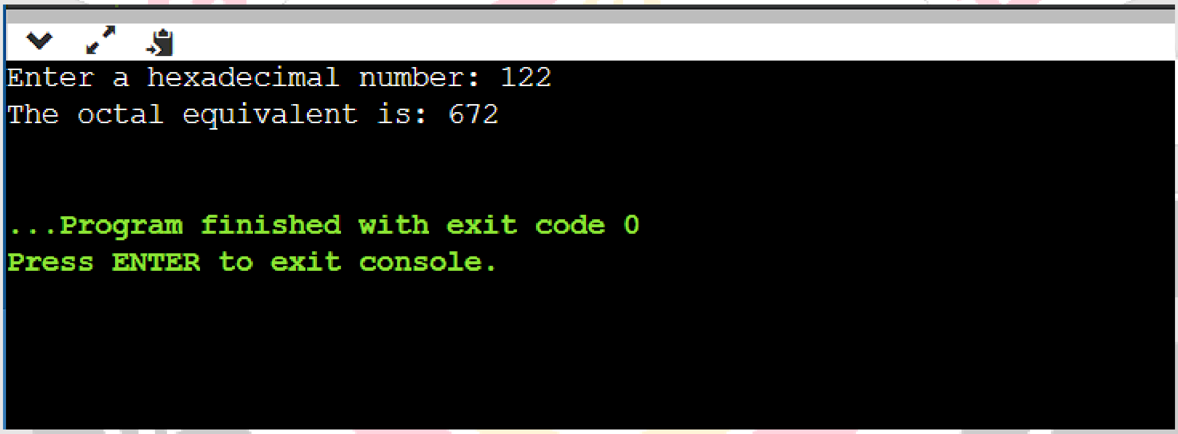
```
Enter a hexadecimal number: 33
Binary equivalent: 00110011


...Program finished with exit code 0
Press ENTER to exit console.
```

71. Write a C program to convert Hexadecimal to Octal number system

```
#include <stdio.h>
#include <string.h>
int main() {
    char hex[20];
```

```
printf("Enter a hexadecimal number: ");
scanf("%s", hex);

long int dec = 0, oct = 0, i = 0;

// Convert hexadecimal to decimal
for (int j = strlen(hex) - 1; j >= 0; j--) {
    int digit;
    if (hex[j] >= '0' && hex[j] <= '9')
        digit = hex[j] - '0';
    else if (hex[j] >= 'A' && hex[j] <= 'F')
        digit = hex[j] - 'A' + 10;
    else if (hex[j] >= 'a' && hex[j] <= 'f')
        digit = hex[j] - 'a' + 10;
    dec += digit * (1 << (4 * i));
    i++;
}

i = 1;
// Convert decimal to octal
while (dec != 0) {
    oct += (dec % 8) * i;
    dec /= 8;
```

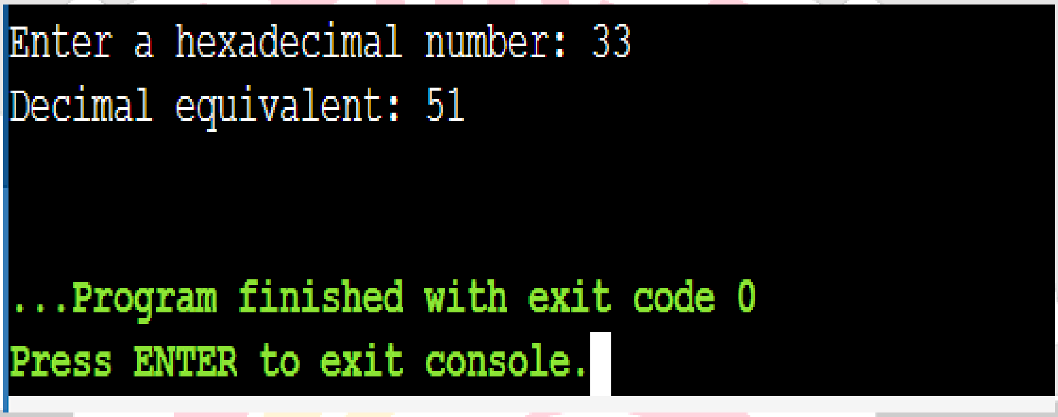
```
        i *= 10;
    }

    printf("The octal equivalent is: %lo\n", oct);

    return 0;
}
//Vinay Kumar
```



```
Enter a hexadecimal number: 122
The octal equivalent is: 672


...Program finished with exit code 0
Press ENTER to exit console.
```

72. Write a C program to convert Hexadecimal to Decimal number system.

```
#include <stdio.h>
#include <math.h>
int main() {
    char hexNum[10];
    int decimalNum = 0;
    int i, j, len;

    printf("Enter a hexadecimal number: ");
    scanf("%s", hexNum);

    // Find the length of the hexadecimal number
    len = strlen(hexNum);

    // Iterate through the hexadecimal number in reverse order
    for (i = 0; hexNum[i] != '\0'; i++) {
        len--;
        if (hexNum[i] >= '0' && hexNum[i] <= '9')
            j = hexNum[i] - '0';
        else if (hexNum[i] >= 'a' && hexNum[i] <= 'f')
            j = hexNum[i] - 'a' + 10;
        else if (hexNum[i] >= 'A' && hexNum[i] <= 'F')
            j = hexNum[i] - 'A' + 10;

        decimalNum += j * pow(16, len);
    }
    printf("Decimal equivalent: %d\n", decimalNum);

    return 0;
}
//Vinay Kumar
```



```
Enter a hexadecimal number: 33
Decimal equivalent: 51


...Program finished with exit code 0
Press ENTER to exit console.
```

Pattern Exercises

1. Star pattern programs - Write a C program to print the given star patterns.

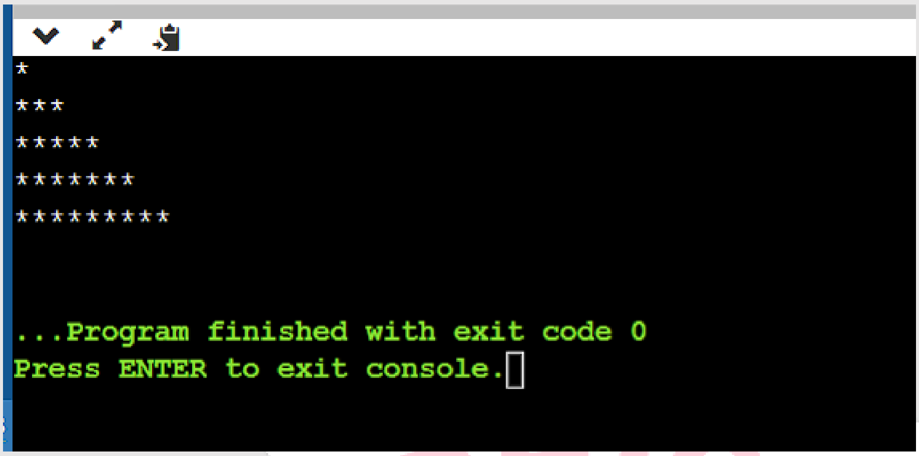
```
*
***
*****
```

****

****

Pyramid Star Pattern

```
#include <stdio.h>

void printStarPattern(int n) {
    for(int i = 1; i <= n; i++) {
        for(int j = 1; j <= 2*i-1; j++) {
            printf("*");
        }
        printf("\n");
    }
}

int main() {
    int n = 5; // Change this value to adjust the number of rows
    printStarPattern(n);
    return 0;
}
//Vinay Kumar
```



```
*
***
*****
*******
*********


...Program finished with exit code 0
Press ENTER to exit console.
```

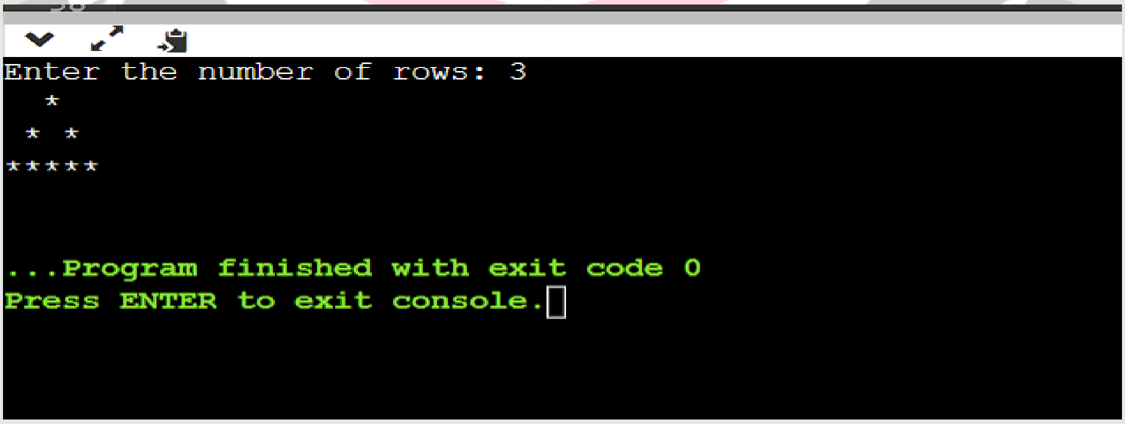
```
2.  *
   * *
  * *
```

```
 * *
*********
```

Hollow Pyramid Star Pattern

```
#include <stdio.h>

void printHollowPyramid(int n) {
    int i, j;
    // Print upper part of the pyramid
    for(i = 1; i <= n; i++) {
        for(j = i; j < n; j++) {
            printf(" ");
        }
        for(j = 1; j <= 2*i-1; j++) {
            if(j == 1 || j == 2*i-1 || i == n) {
                printf("*");
            }
            else {
                printf(" ");
            }
        }
        printf("\n");
    }
}
```

```
int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    if(n <= 0) {
        printf("Invalid input. Please enter a positive integer.\n");
        return 1;
    }

    printHollowPyramid(n);

    return 0;
}
```



```
Enter the number of rows: 3
  *
 * *
*****


...Program finished with exit code 0
Press ENTER to exit console.
```

//Vinay Kumar

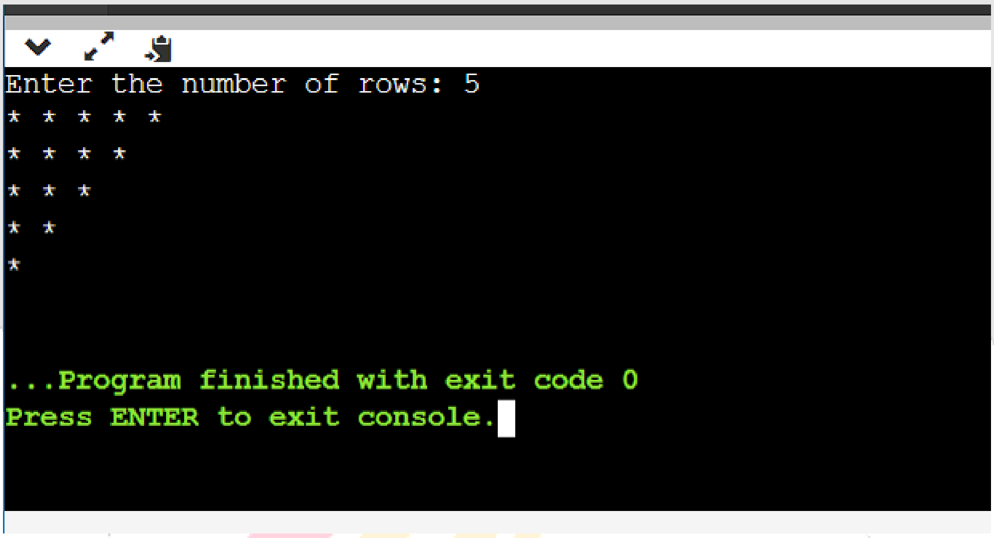
****

```
*******

*****

***

*
```

3.Inverted Pyramid Star Pattern

```
#include <stdio.h>

int main() {
    int i, j, rows;

    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    for(i = rows; i >= 1; --i) {
        for(j = 1; j <= i; ++j) {
            printf("* ");
        }
        printf("\n");
    }

    return 0;
}
```

//Vinay Kumar



```
Enter the number of rows: 5
* * * * *
* * * *
* * *
* *
*


...Program finished with exit code 0
Press ENTER to exit console.
```

****

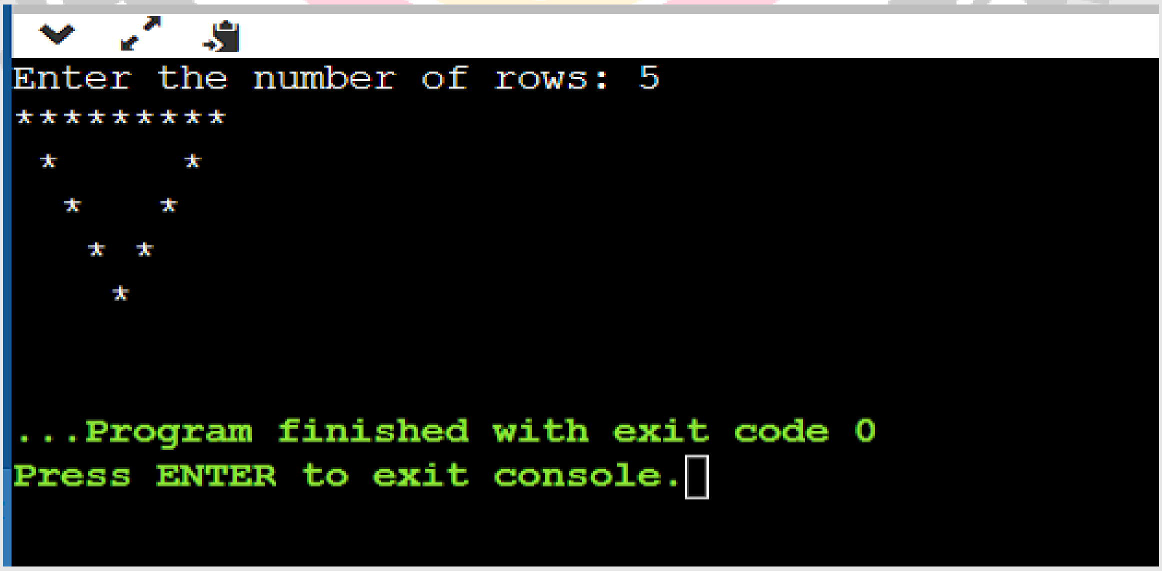
* *

```
* *

* *

*
```

4.Hollow Inverted Pyramid Star Pattern

```
#include <stdio.h>

int main() {
    int rows, i, j;

    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    for(i = 1; i <= rows; i++) {
        // Print spaces
        for(j = 1; j < i; j++) {
            printf(" ");
        }

        // Print stars for the first row or the last row
        if(i == 1 || i == rows) {
            for(j = 1; j <= (2 * (rows - i) + 1); j++) {
                printf("*");
            }
        } else { // Print stars for other rows
```

```
            printf("*");

            for(j = 1; j < (2 * (rows - i)); j++) {
                printf(" ");
            }

            printf("*");
        }

        printf("\n");
    }

    return 0;
}
```



```
Enter the number of rows: 5
*********
 *     *
  *   *
   * *
    *


...Program finished with exit code 0
Press ENTER to exit console.
```

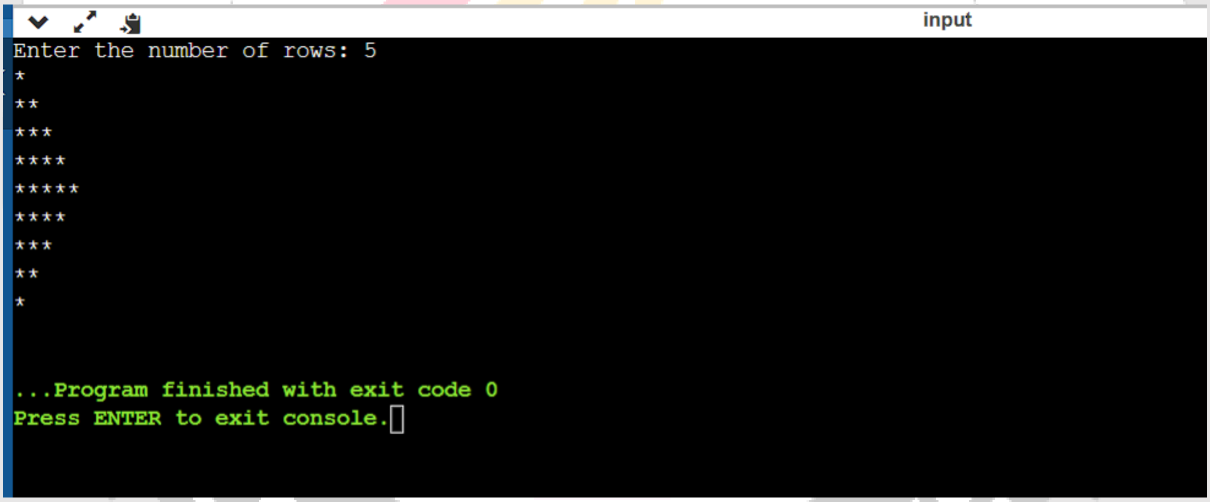
//Vinay Kumar

```
*****
*
**
***
****


****
***
**
*
```

5.Half Diamond Star Pattern

```
#include <stdio.h>


int main() {
    int i, j, rows;


    printf("Enter the number of rows: ");
    scanf("%d", &rows);


    // Print upper half of the diamond
    for(i = 1; i <= rows; i++) {
        for(j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }


    // Print lower half of the diamond
    for(i = rows-1; i >= 1; i--) {
        for(j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }

    return 0;
}
```

//Vinay Kumar



```
Enter the number of rows: 5
*
**
***
****
*****
****
***
**
*


...Program finished with exit code 0
Press ENTER to exit console.
```

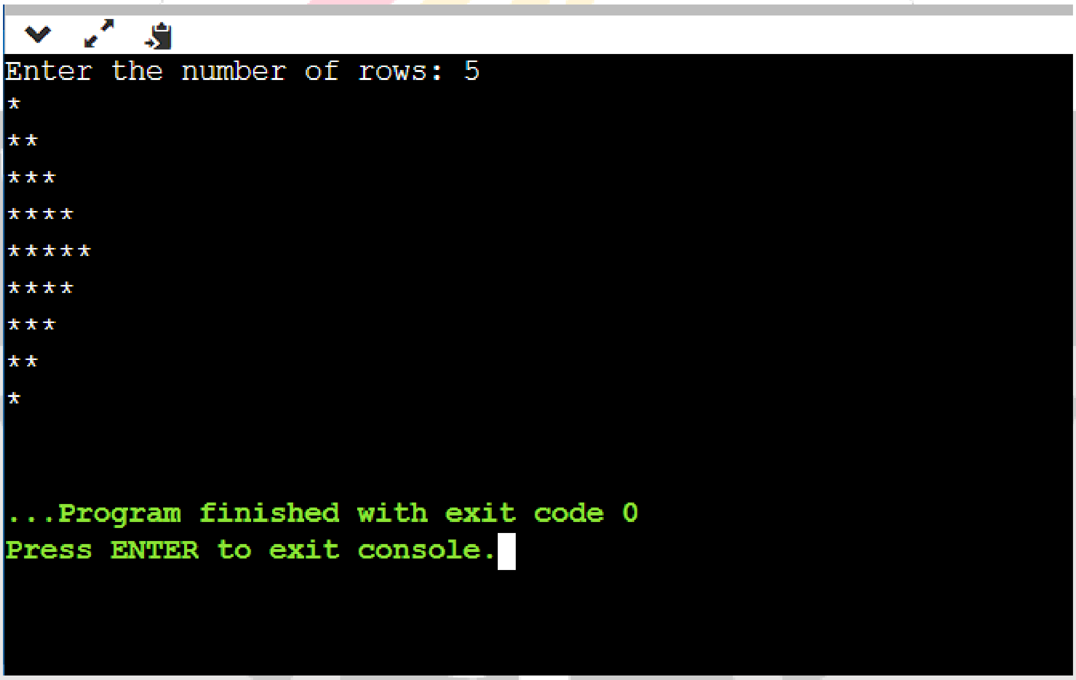
```
    *
   **
  ***
 ****
*****
 ****
  ***
   **
    *
```

6.Mirrored Half Diamond Star Pattern

```
#include <stdio.h>

void printHalfDiamond(int n) {
int i, j;

// Upper half of the pattern
for(i = 1; i <= n; i++) {
for(j = 1; j <= i; j++) {
printf("*");
}
printf("\n");
}

// Lower half of the pattern
for(i = n-1; i >= 1; i--) {
for(j = 1; j <= i; j++) {
printf("*");
}
printf("\n");
}
}

int main() {
```

```
int n;

// Get user input for the number of rows
printf("Enter the number of rows: ");
scanf("%d", &n);

// Call the function to print the pattern
printHalfDiamond(n);

return 0;
}
```

//Vinay Kumar



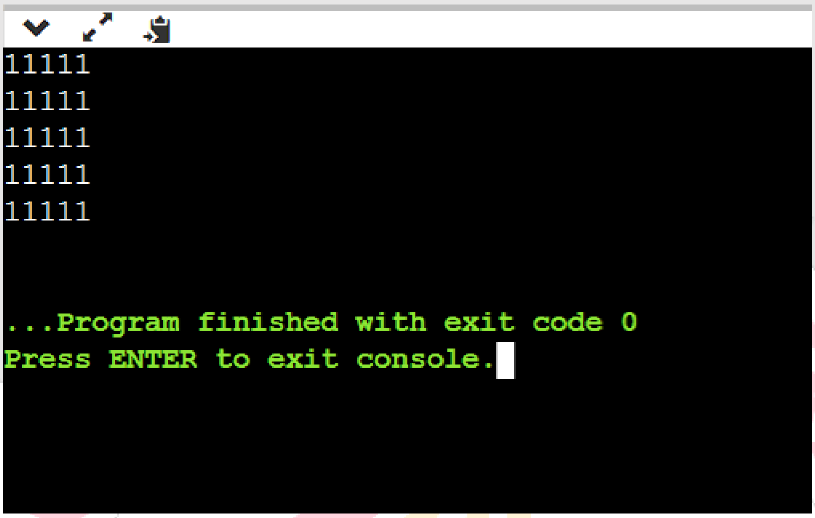
```
Enter the number of rows: 5
*
**
***
****
*****
****
***
**
*


...Program finished with exit code 0
Press ENTER to exit console.
```

2. Number pattern programs - Write a C program to print the given number patterns

Square number patterns

```
11111
11111
11111
11111
11111
```

```
#include <stdio.h>

int main() {
   int i, j;

   for(i = 0; i < 5; i++) {
      for(j = 0; j < 5; j++) {
         printf("1");
      }
      printf("\n");
   }

   return 0;
}
//Vinay Kumar
```



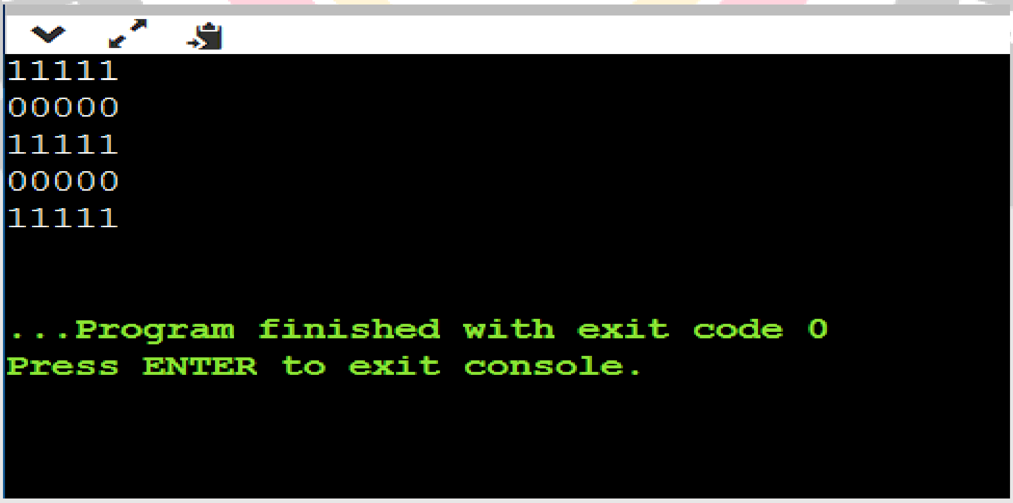
```
11111
11111
11111
11111
11111


...Program finished with exit code 0
Press ENTER to exit console.
```

Number pattern 1

```
11111
00000
11111
00000
11111
```

```
#include <stdio.h>

int main() {
    int i, j;

    for (i = 1; i <= 5; i++) {
        for (j = 1; j <= 5; j++) {
            if (i % 2 == 1) {
                printf("1");
            } else {
                printf("0");
            }
        }
        printf("\n");
    }

    return 0;
}
```

//Vinay Kumar



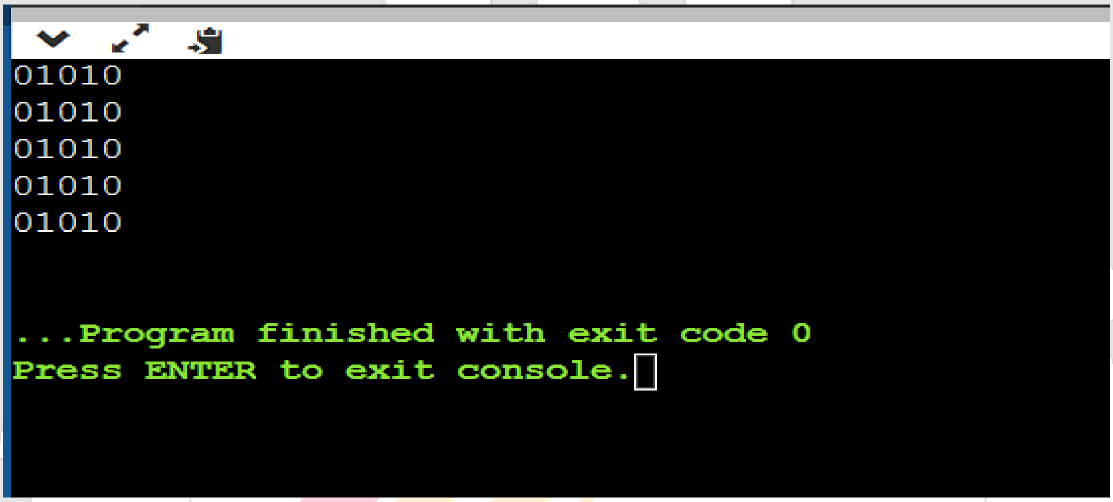
```
11111
00000
11111
00000
11111


...Program finished with exit code 0
Press ENTER to exit console.
```

Number pattern 2

```
01010

01010

01010

01010

01010
```

```
#include <stdio.h>

int main() {
    int i, j;

    for(i = 1; i <= 5; i++) {
        for(j = 1; j <= 5; j++) {
            if(j % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }
        printf("\n");
    }

    return 0;
}
```

//Vinay Kumar

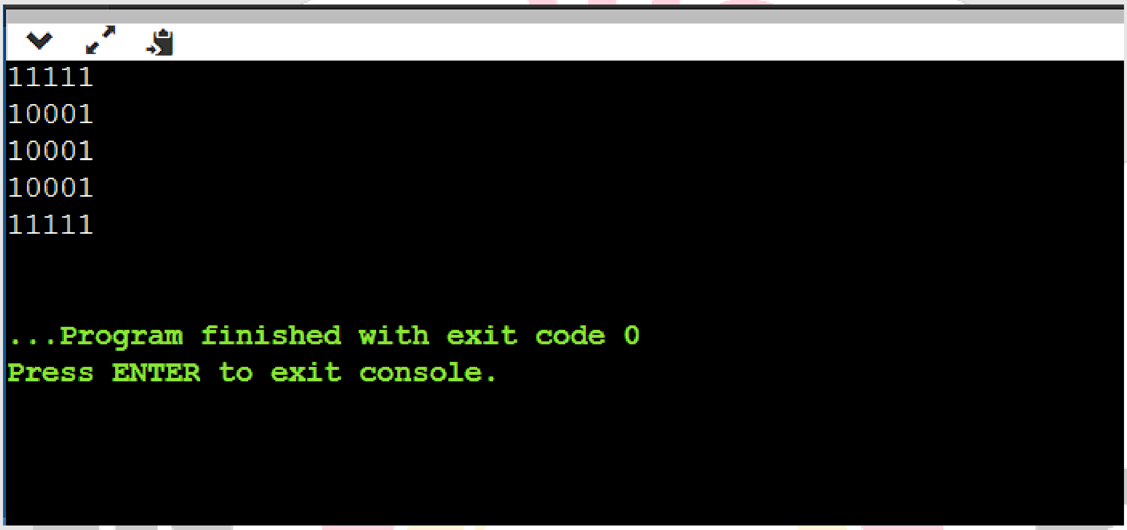


```
01010
01010
01010
01010
01010


...Program finished with exit code 0
Press ENTER to exit console.
```

Number pattern 3

```
11111
10001
10001
10001
11111
```

```
#include <stdio.h>

int main() {
    int n = 5; // Number of rows and columns

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if (i == 0 || i == n - 1 || j == 0 || j == n - 1)
                printf("1");
            else
                printf("0");
        }
        printf("\n");
    }
    return 0;
}
//Vinay Kumar
```

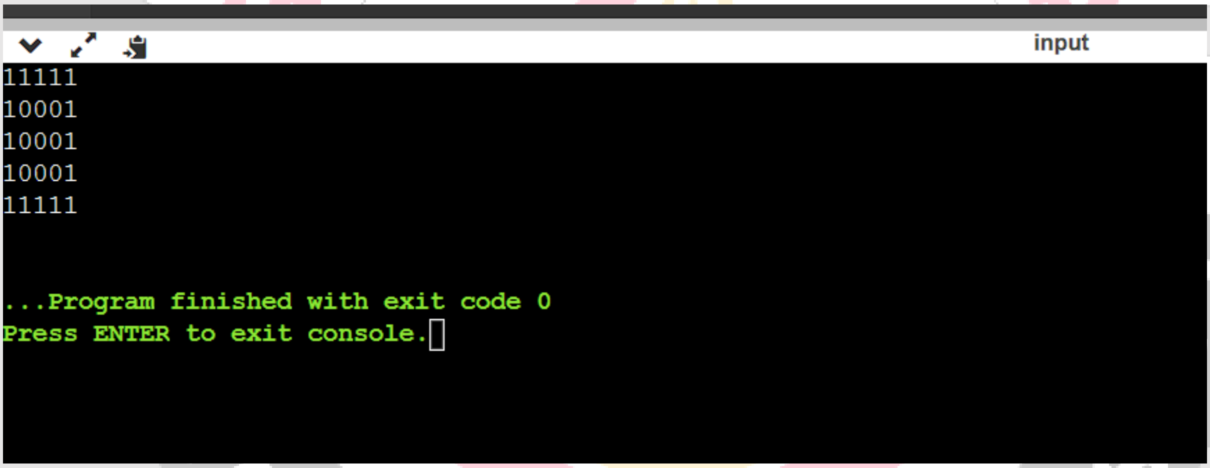


```
11111
10001
10001
10001
11111


...Program finished with exit code 0
Press ENTER to exit console.
```

Number pattern 4

```
11111
11111
11011
11111
11111
```

```
#include <stdio.h>

int main() {
    int n = 5; // Size of the pattern (5x5 in this case)

    for(int i = 0; i < n; i++) {
        for(int j = 0; j < n; j++) {
            if (i == n/2 && j == n/2) { // If we're at the center,
print 0
                printf("0");
            } else {
                printf("1");
            }
        }
        printf("\n"); // Move to the next line after each row
    }

    return 0;
}
//Vinay Kumar
```



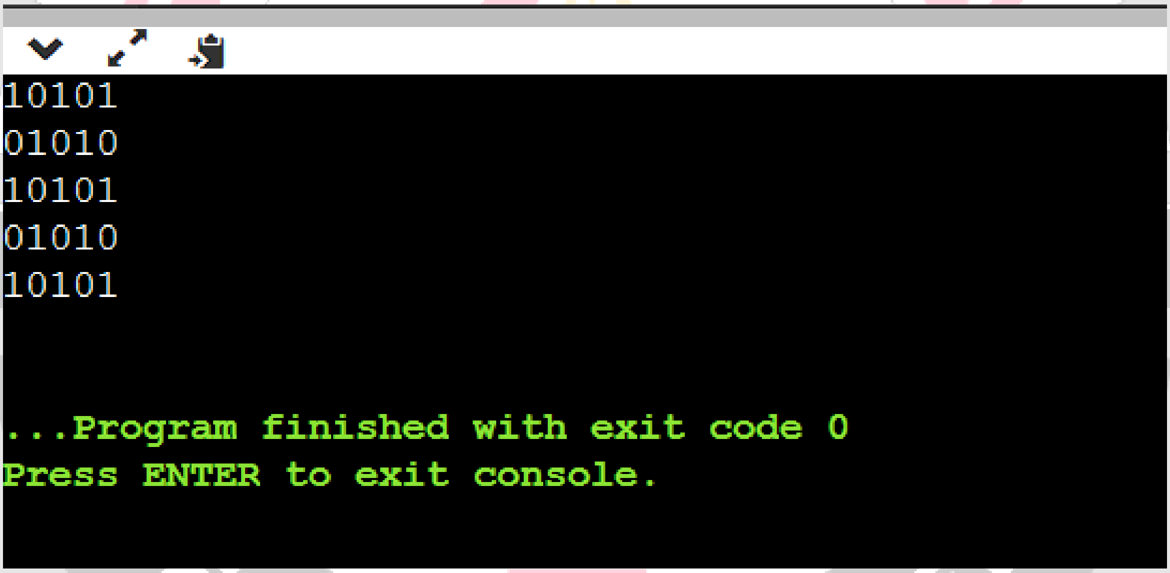
```
11111
10001
10001
10001
11111


...Program finished with exit code 0
Press ENTER to exit console.
```

Number pattern 5

```
10101
01010
10101
01010
10101
```

```
#include <stdio.h>

int main() {
    int rows = 5;
    int cols = 5;

    for(int i = 0; i < rows; i++) {
        for(int j = 0; j < cols; j++) {
            if((i+j) % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }
        printf("\n");
    }

    return 0;
}
```

//Vinay Kumar

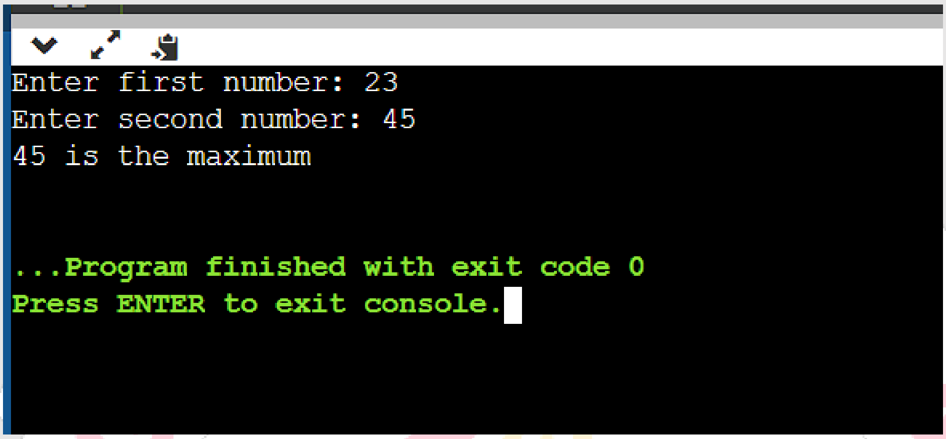


```
10101
01010
10101
01010
10101


...Program finished with exit code 0
Press ENTER to exit console.
```

If…Else Exercises

1. **Write a C program to find maximum between two numbers.**

```
#include <stdio.h>

int main() {
    int num1, num2;

    // Input the two numbers
    printf("Enter first number: ");
    scanf("%d", &num1);

    printf("Enter second number: ");
    scanf("%d", &num2);

    // Compare the numbers
    if (num1 > num2) {
        printf("%d is the maximum\n", num1);
    } else {
        printf("%d is the maximum\n", num2);
    }

    return 0;
    }
//Vinay Kumar
```

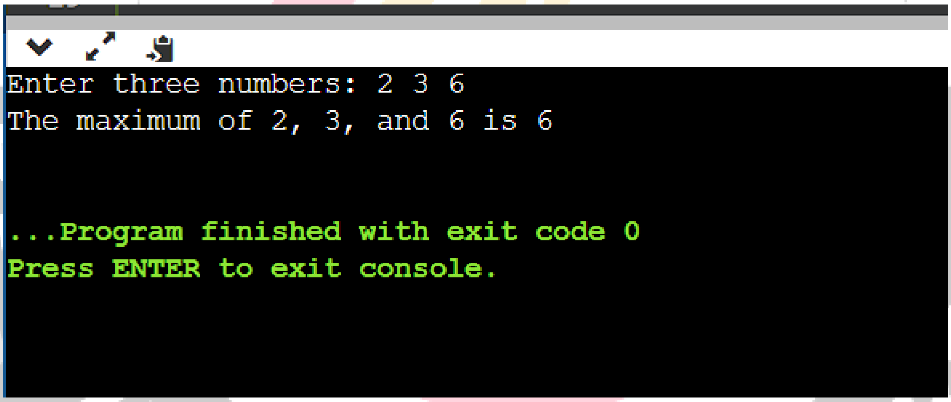


```
Enter first number: 23
Enter second number: 45
45 is the maximum


...Program finished with exit code 0
Press ENTER to exit console.
```

2. Write a C program to find maximum between three numbers.

```
#include <stdio.h>

int findMax(int a, int b, int c) {
    int max = a;

    if (b > max) {
        max = b;
    }

    if (c > max) {
        max = c;
    }

    return max;
}

int main() {
    int num1, num2, num3;
    printf("Enter three numbers: ");
    scanf("%d %d %d", &num1, &num2, &num3);

    int max = findMax(num1, num2, num3);

    printf("The maximum of %d, %d, and %d is %d\n", 
num1, num2, num3, max);

    return 0;
}
```

//Vinay Kumar

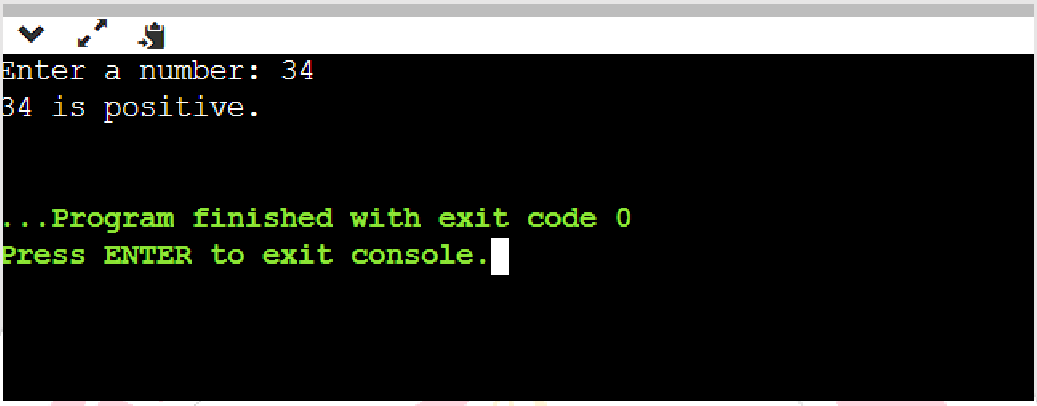


```
Enter three numbers: 2 3 6
The maximum of 2, 3, and 6 is 6


...Program finished with exit code 0
Press ENTER to exit console.
```

3. Write a C program to check whether a number is negative, positive or zero.

```
#include <stdio.h>

int main() {
    int num;

    // Read the number from the user
    printf("Enter a number: ");
    scanf("%d", &num);

    if (num > 0) {
        printf("%d is positive.\n", num);
    } else if (num < 0) {
        printf("%d is negative.\n", num);
    } else {
        printf("%d is zero.\n", num);
    }

    return 0;
}
```

//Vinay Kumar

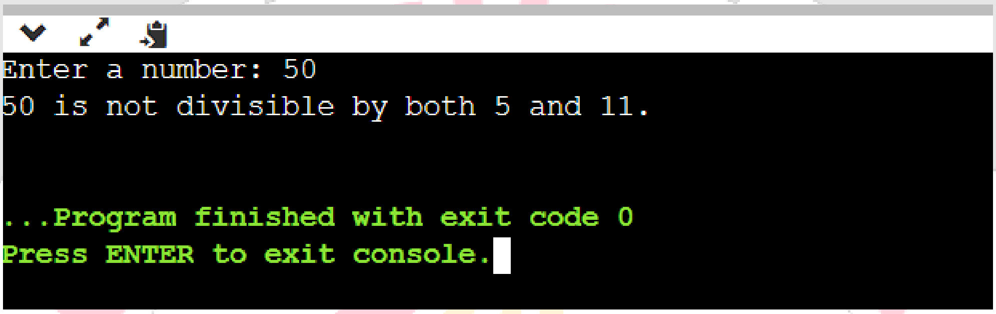


```
Enter a number: 34
34 is positive.


...Program finished with exit code 0
Press ENTER to exit console.
```

4. **Write a C program to check whether a number is divisible by 5 and 11 or not.**

```
#include <stdio.h>

int main() {
    int num;

    // Read input from user
    printf("Enter a number: ");
    scanf("%d", &num);

    // Check if the number is divisible by both 5 and 11
    if (num % 5 == 0 && num % 11 == 0) {
        printf("%d is divisible by both 5 and 11.\n", num);
    } else {
        printf("%d is not divisible by both 5 and 11.\n", num);
    }

    return 0;
}
```

//Vinay Kumar

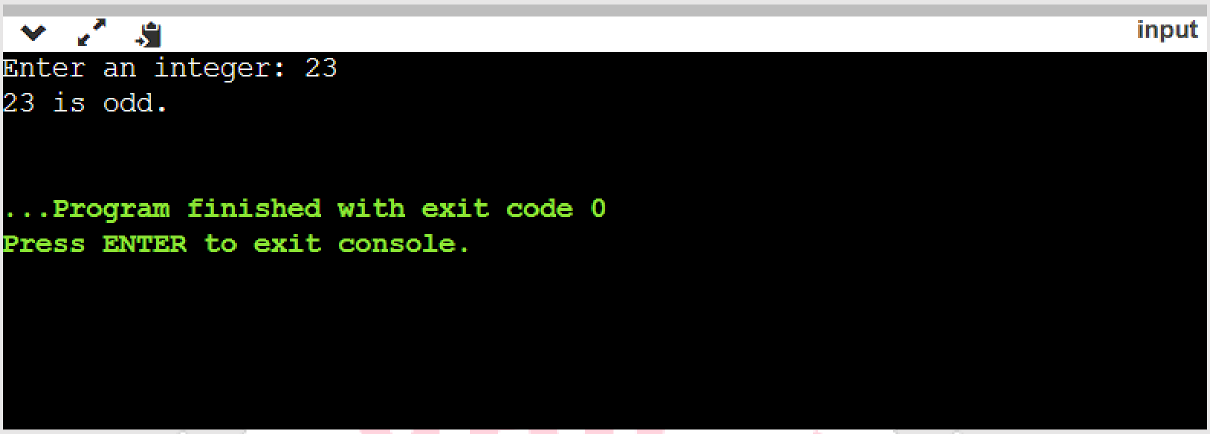


```
Enter a number: 50
50 is not divisible by both 5 and 11.


...Program finished with exit code 0
Press ENTER to exit console.
```

5. Write a C program to check whether a number is even or odd.

```
#include <stdio.h>

int main() {
    int num;

    // Prompt the user to enter a number
    printf("Enter an integer: ");
    scanf("%d", &num);

    // Check if the number is even or odd
    if (num % 2 == 0) {
        printf("%d is even.\n", num);
    } else {
        printf("%d is odd.\n", num);
    }
    return 0;
}
```

//Vinay Kumar



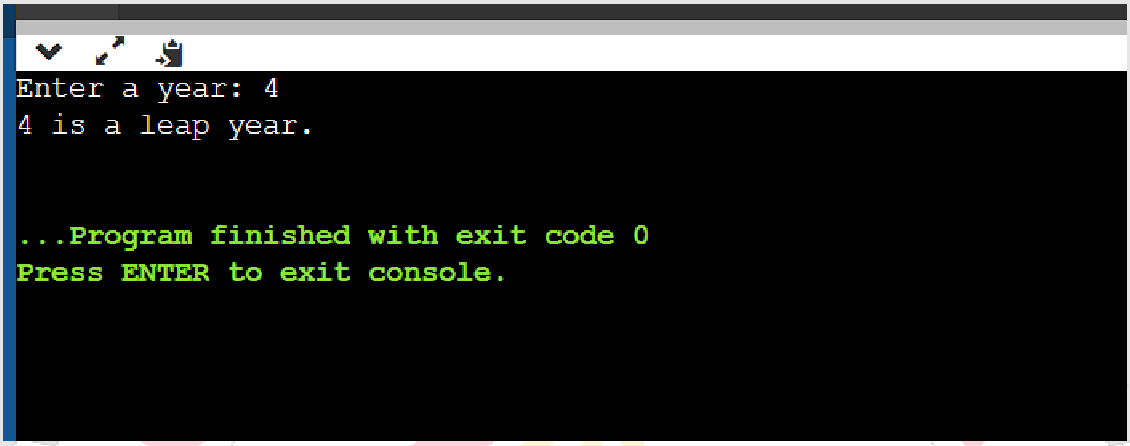
```
Enter an integer: 23
23 is odd.


...Program finished with exit code 0
Press ENTER to exit console.
```

6. **Write a C program to check whether a year is leap year or not.**

```
 #include <stdio.h>

int main() {
   int year;

   // Input year from user
   printf("Enter a year: ");
   scanf("%d", &year);

   // Check if the year is a leap year
   if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
      printf("%d is a leap year.\n", year);
   } else {
      printf("%d is not a leap year.\n", year);
   }
```

```
        return 0;
    }
```

//Vinay Kumar



```
Enter a year: 4
4 is a leap year.


...Program finished with exit code 0
Press ENTER to exit console.
```

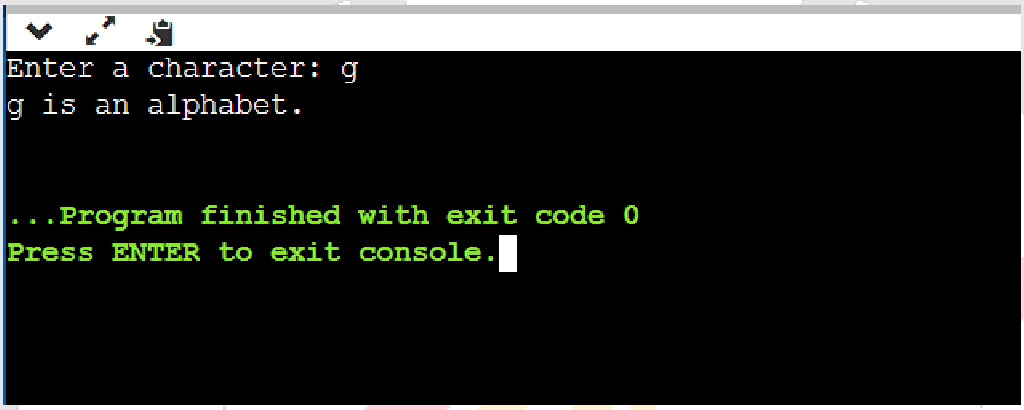
7.

Write a C program to check whether a character is alphabet or not #include <stdio.h>

```
int main() {
   char ch;

   // Input character from user
   printf("Enter a character: ");
   scanf("%c", &ch);

   // Check if the character is an alphabet
   if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {
      printf("%c is an alphabet.\n", ch);
   } else {
      printf("%c is not an alphabet.\n", ch);
   }
```

```
    return 0;
}
```

//Vinay Kumar

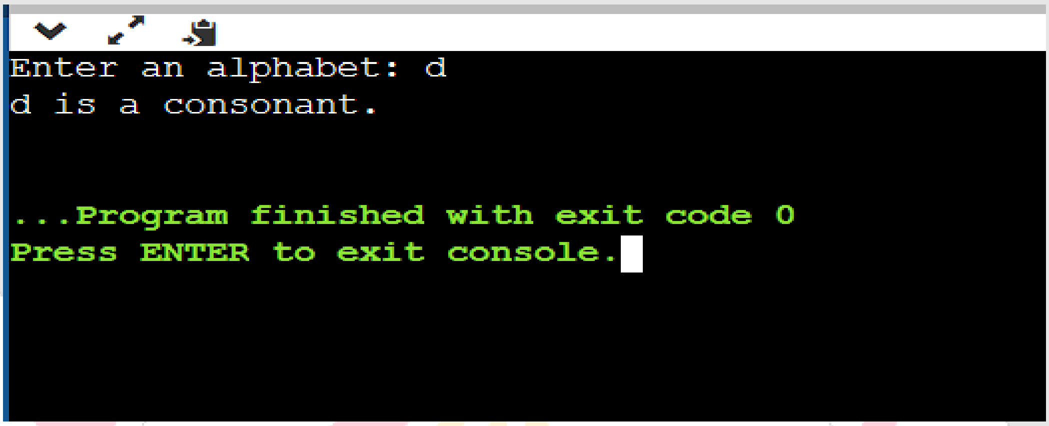


```
Enter a character: g
g is an alphabet.


...Program finished with exit code 0
Press ENTER to exit console.
```

7. Write a C program to input any alphabet and check whether it is vowel or consonant

```
#include <stdio.h>

int main() {
    char ch;

    printf("Enter an alphabet: ");
    scanf("%c", &ch);

    // Using switch case to check if it's a vowel or consonant
    switch(ch) {
        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u':
        case 'A':
        case 'E':
        case 'I':
        case 'O':
        case 'U':
            printf("%c is a vowel.\n", ch);
            break;
        default:
            printf("%c is a consonant.\n", ch);
    }

    return 0;
}
```

//Vinay Kumar



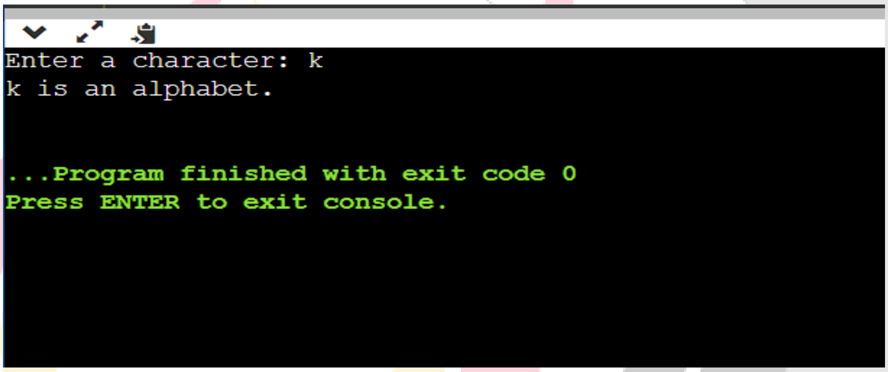
```
Enter an alphabet: d
d is a consonant.


...Program finished with exit code 0
Press ENTER to exit console.
```

9. Write a C program to input any character and check whether it is alphabet, digit or special character.

```
#include <stdio.h>

int main() {
   char character;

   // Ask user to enter a character
   printf("Enter a character: ");
   scanf("%c", &character);

   // Check if the character is an alphabet
   if ((character >= 'a' && character <= 'z') || (character >= 'A' && character <= 'Z')) {
      printf("%c is an alphabet.\n", character);
   }
```

```
    // Check if the character is a digit
    else if (character >= '0' && character <= '9') {
        printf("%c is a digit.\n", character);
    }
    // If it's not an alphabet or a digit, it's a special character
    else {
        printf("%c is a special character.\n", character);
    }

    return 0;
}
```

//Vinay Kumar



```
Enter a character: k
k is an alphabet.


...Program finished with exit code 0
Press ENTER to exit console.
```

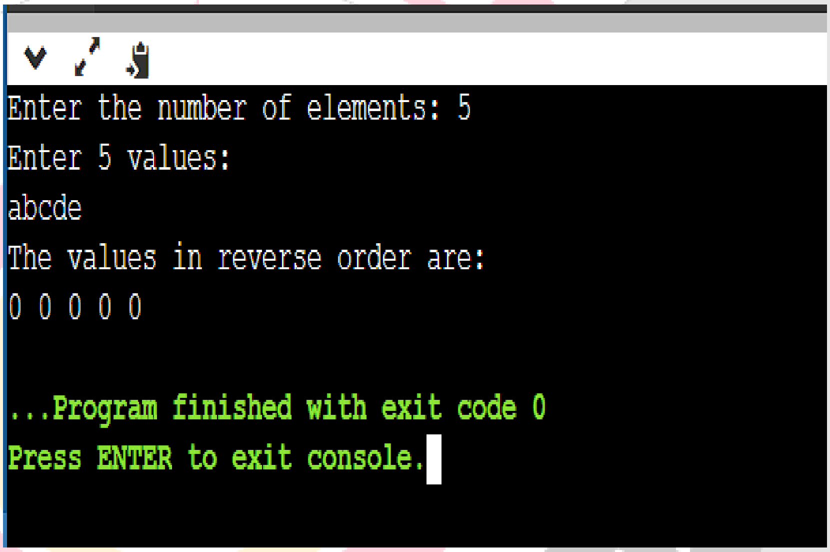
////Arrays

1. **Write a program in C to read n number of values in an array and display them in reverse order.**

```
#include <stdio.h>

int main() {
    int n;

    printf("Enter the number of elements: ");
    scanf("%d", &n);

    int arr[n];
```

```
        printf("Enter %d values:\n", n);
        for(int i = 0; i < n; i++) {
            scanf("%d", &arr[i]);
        }

        printf("The values in reverse order are:\n");
        for(int i = n - 1; i >= 0; i--) {
            printf("%d ", arr[i]);
        }

        return 0;
    }
//Vinay Kumar
```

Input:

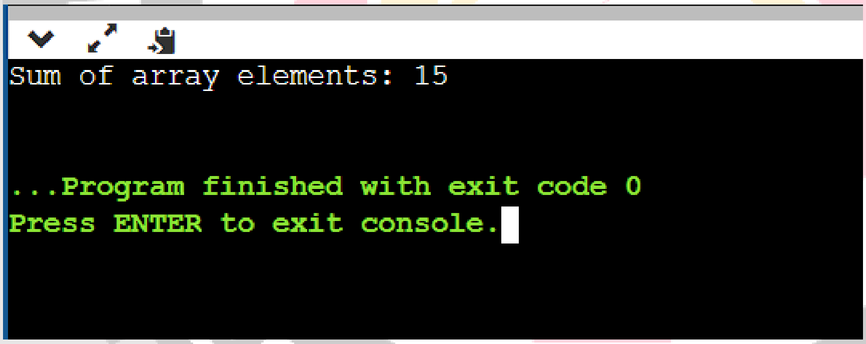


```
Enter the number of elements: 5
Enter 5 values:
abcde
The values in reverse order are:
0 0 0 0 0

...Program finished with exit code 0
Press ENTER to exit console.
```

2. Write a program in C to find the sum of all elements of the array.

```
#include <stdio.h>


int main() {
    int arr[5] = {1, 2, 3, 4, 5}; // You can change the array elements here
    int sum = 0;
    // Loop through the array and add each element to the sum
    for(int i = 0; i < 5; i++) {
        sum += arr[i];
    }

    printf("Sum of array elements: %d\n", sum);

    return 0;
}
//Vinay Kumar
```

Input:

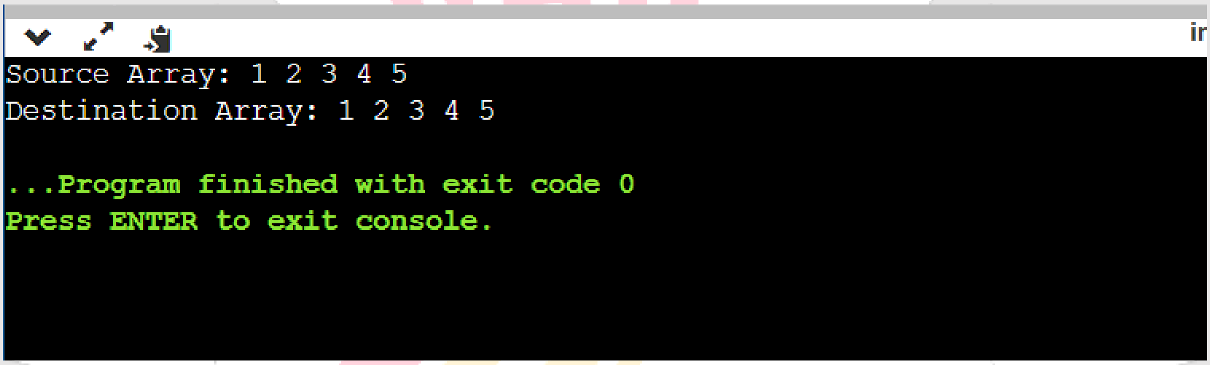


```
Sum of array elements: 15


...Program finished with exit code 0
Press ENTER to exit console.
```

3. Write a program in C to copy the elements of one array into another array.

```
#include <stdio.h>

void copyArray(int source[], int destination[], int size) {
    for (int i = 0; i < size; i++) {
        destination[i] = source[i];
    }
}

int main() {
    int sourceArray[] = {1, 2, 3, 4, 5};
    int destinationArray[5]; // Make sure the destination array has enough space

    int size = sizeof(sourceArray) / sizeof(sourceArray[0]);

    copyArray(sourceArray, destinationArray, size);

    printf("Source Array: ");
    for (int i = 0; i < size; i++) {
        printf("%d ", sourceArray[i]);
    }

    printf("\n");

    printf("Destination Array: ");
    for (int i = 0; i < size; i++) {
        printf("%d ", destinationArray[i]);
    }
    return 0;
}
//Vinay Kumar
```

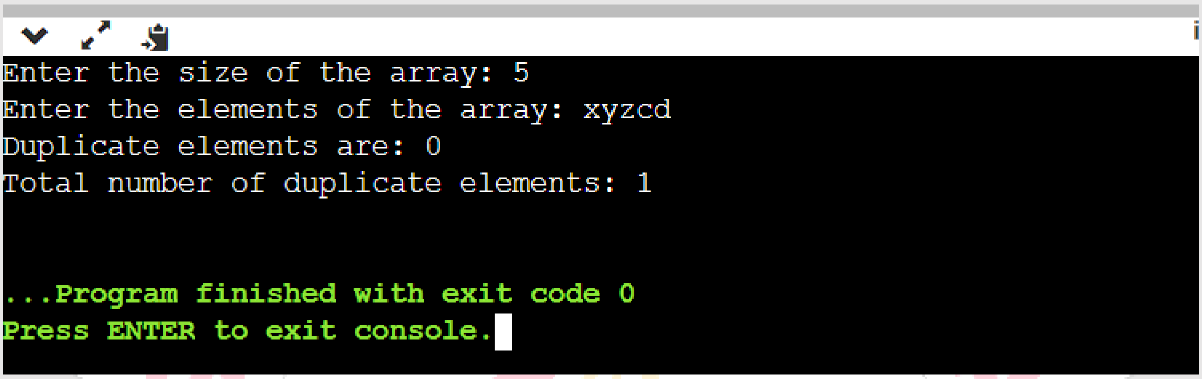
Input:



```
Source Array: 1 2 3 4 5
Destination Array: 1 2 3 4 5

...Program finished with exit code 0
Press ENTER to exit console.
```

4. **Write a program in C to count the total number of duplicate elements in an array.**

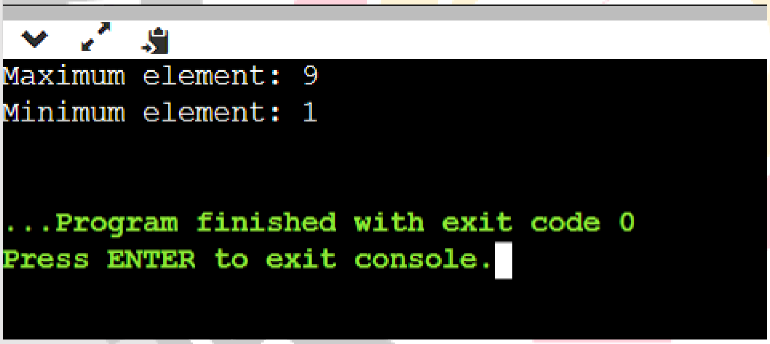
```
#include <stdio.h>


int main() {
    int arr[100], freq[100], n, count = 0;

    // Input the size of the array
    printf("Enter the size of the array: ");
    scanf("%d", &n);

    // Input the elements of the array
    printf("Enter the elements of the array: ");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        freq[i] = -1; // Initialize frequency array with -1
    }

    // Count frequency of each element
    for (int i = 0; i < n; i++) {
        int freqCount = 1;
        for (int j = i+1; j < n; j++) {
            if (arr[i] == arr[j]) {
                freqCount++;
                freq[j] = 0; // Mark the duplicate element
            }
        }
        if (freq[i] != 0) {
            freq[i] = freqCount;
        }
    }

    // Count and display duplicate elements
    printf("Duplicate elements are: ");
    for (int i = 0; i < n; i++) {
        if (freq[i] > 1) {
            printf("%d ", arr[i]);
            count++;
        }
    }

    if (count == 0) {
        printf("None\n");
    } else {
        printf("\nTotal number of duplicate elements: %d\n",
count);
    }
    return 0;
}
//Vinay Kumar
```



```
Enter the size of the array: 5
Enter the elements of the array: xyzcd
Duplicate elements are: 0
Total number of duplicate elements: 1


...Program finished with exit code 0
Press ENTER to exit console.
```

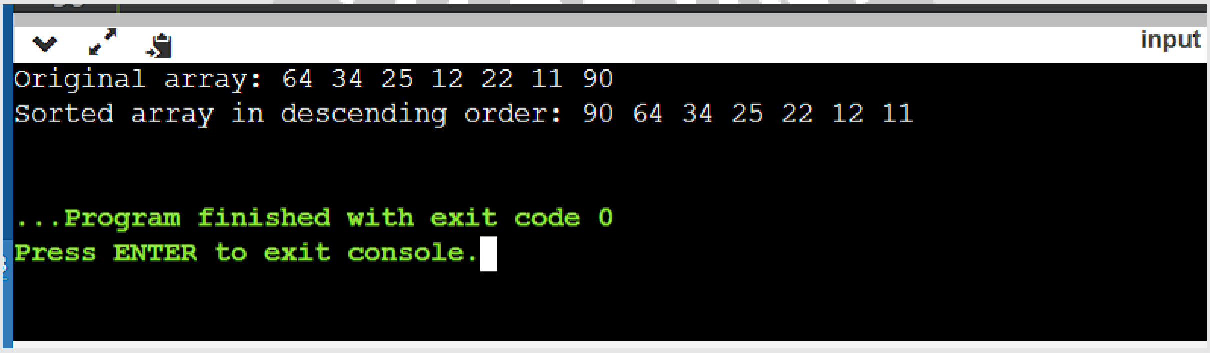
```
#include <stdio.h>

void findMinMax(int arr[], int size, int *max, int *min) {
    *max = *min = arr[0]; // Initialize max and min with the first element of the array

    for (int i = 1; i < size; i++) {
        if (arr[i] > *max) {
            *max = arr[i];
        }
        if (arr[i] < *min) {
            *min = arr[i];
        }
    }
}

int main() {
    int arr[] = {3, 1, 7, 9, 4, 2, 6, 8, 5};
    int size = sizeof(arr) / sizeof(arr[0]);
    int max, min;
    findMinMax(arr, size, &max, &min);

    printf("Maximum element: %d\n", max);
    printf("Minimum element: %d\n", min);

    return 0;
}
```

//Vinay Kumar



```
Maximum element: 9
Minimum element: 1


...Program finished with exit code 0
Press ENTER to exit console.
```

```
#include <stdio.h>

void bubbleSort(int arr[], int n) {
    for (int i = 0; i < n-1; i++) {
        for (int j = 0; j < n-i-1; j++) {
            if (arr[j] < arr[j+1]) {
                // Swap arr[j] and arr[j+1]
                int temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
            }
        }
    }
}

int main() {
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int n = sizeof(arr)/sizeof(arr[0]);
    printf("Original array: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");

    bubbleSort(arr, n);

    printf("Sorted array in descending order: ");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");

    return 0;
}
//Vinay Kumar
```

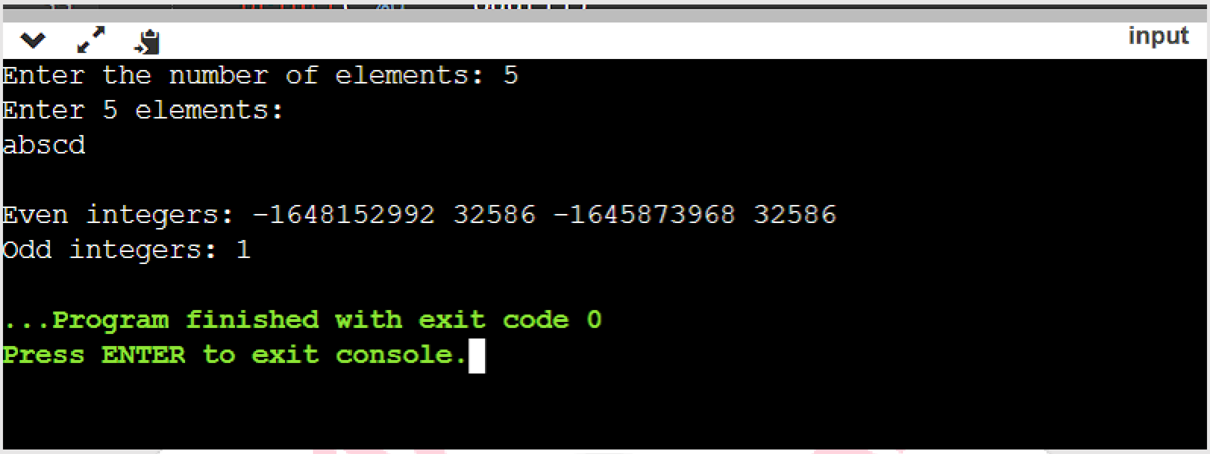


```
Original array: 64 34 25 12 22 11 90
Sorted array in descending order: 90 64 34 25 22 12 11


...Program finished with exit code 0
Press ENTER to exit console.
```

7. Write a program in C to separate odd and even integers into separate arrays.

```
#include <stdio.h>

int main() {
    int arr[100], even[100], odd[100];
    int n, even_count = 0, odd_count = 0;

    // Input the number of elements in the array
    printf("Enter the number of elements: ");
    scanf("%d", &n);

    // Input the elements of the array
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    // Separate odd and even integers
    for (int i = 0; i < n; i++) {
        if (arr[i] % 2 == 0) {
            even[even_count++] = arr[i];
        } else {
            odd[odd_count++] = arr[i];
        }
    }

    // Print even integers
    printf("\nEven integers: ");
    for (int i = 0; i < even_count; i++) {
        printf("%d ", even[i]);
    }

    // Print odd integers
    printf("\nOdd integers: ");
    for (int i = 0; i < odd_count; i++) {
        printf("%d ", odd[i]);
    }

    return 0;
}
```

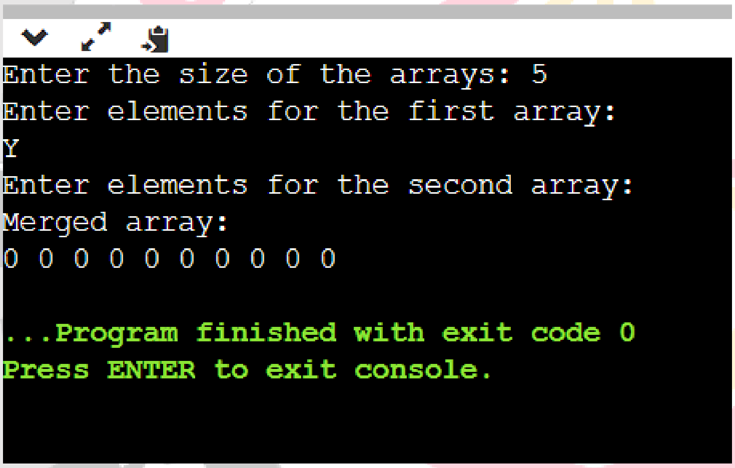
//Vinay Kumar



```
Enter the number of elements: 5
Enter 5 elements:
abscd

Even integers: -1648152992 32586 -1645873968 32586
Odd integers: 1

...Program finished with exit code 0
Press ENTER to exit console.
```

8. Write a program in C to merge two arrays of the same size sorted in descending/ascending order.

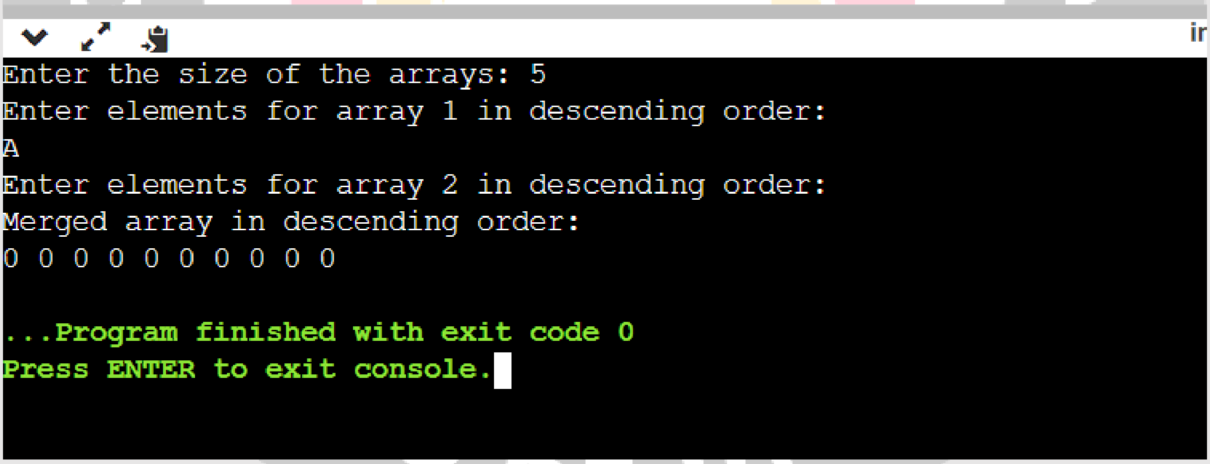
```
#include <stdio.h>


void mergeArrays(int arr1[], int arr2[], int n, int result[]) {
    int i = 0, j = 0, k = 0;
    while (i < n && j < n) {
        if (arr1[i] <= arr2[j]) {
            result[k++] = arr1[i++];
        } else {
            result[k++] = arr2[j++];
        }
    }
    while (i < n) {
        result[k++] = arr1[i++];
    }
    while (j < n) {
        result[k++] = arr2[j++];
    }
}

int main() {
    int n;
    printf("Enter the size of the arrays: ");
    scanf("%d", &n);

    int arr1[n], arr2[n], result[2*n];

    printf("Enter elements for the first array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr1[i]);
    }

    printf("Enter elements for the second array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr2[i]);
    }

    // Assuming both arrays are already sorted in ascending
order.
    mergeArrays(arr1, arr2, n, result);
    printf("Merged array:\n");
    for (int i = 0; i < 2*n; i++) {
        printf("%d ", result[i]);
    }

    return 0;
}
//Vinay Kumar
```



```
Enter the size of the arrays: 5
Enter elements for the first array:
Y
Enter elements for the second array:
Merged array:
0 0 0 0 0 0 0 0 0 0

...Program finished with exit code 0
Press ENTER to exit console.
```

9. Write a program in C to merge two arrays of the same size sorted in descending order.

```
#include <stdio.h>

void mergeArrays(int arr1[], int arr2[], int size, int result[]) {
    int i = 0, j = 0, k = 0;

    while (i < size && j < size) {
        if (arr1[i] >= arr2[j]) {
            result[k++] = arr1[i++];
        } else {
            result[k++] = arr2[j++];
        }
    }

    while (i < size) {
        result[k++] = arr1[i++];
    }

    while (j < size) {
        result[k++] = arr2[j++];
    }
}

int main() {
    int size;
    printf("Enter the size of the arrays: ");
    scanf("%d", &size);

    int arr1[size], arr2[size], result[2*size];

    printf("Enter elements for array 1 in descending order:\n");
    for (int i = 0; i < size; i++) {
        scanf("%d", &arr1[i]);
    }

    printf("Enter elements for array 2 in descending order:\n");
    for (int i = 0; i < size; i++) {
        scanf("%d", &arr2[i]);
    }
    mergeArrays(arr1, arr2, size, result);

    printf("Merged array in descending order:\n");
    for (int i = 0; i < 2*size; i++) {
        printf("%d ", result[i]);
    }

    return 0;
}
//Vinay Kumar
```



```
Enter the size of the arrays: 5
Enter elements for array 1 in descending order:
A
Enter elements for array 2 in descending order:
Merged array in descending order:
0 0 0 0 0 0 0 0 0 0

...Program finished with exit code 0
Press ENTER to exit console.
```

1.