

INTRODUCTION TO C PROGRAMMING



BATCH:

2023-2027

BCA HONOURS (AI & DS)

SUBMITTED BY:

TO:

NIMESH BAHUGUNA

SUBMITTED

Mr RISHI KUMAR

STUDENT ID:

ASTT

PROF.CSIT:

2316012116

GEU

INDEX

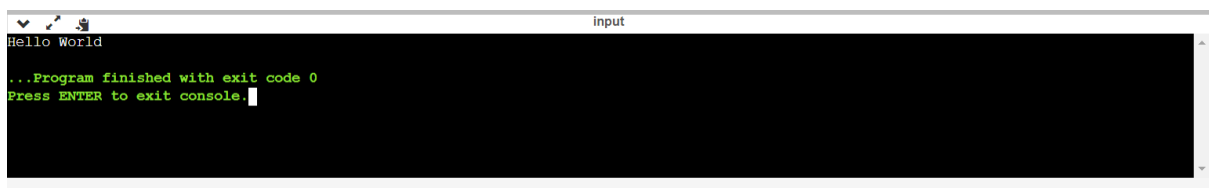
NIMESH BAHUGUNA

1) WAP FOR HELLO WORLD :

```
#include <stdio.h>

int main()
{
    printf("Hello World");

    return 0;
}
```

A screenshot of a terminal window titled 'input'. The terminal has a black background with white text. The first line shows 'Hello World'. The second line shows '...Program finished with exit code 0'. The third line shows 'Press ENTER to exit console.' followed by a cursor. The terminal window has a standard macOS-style title bar with a red, yellow, and green button on the left and a scroll bar on the right.

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

2) WAP TO ADD TWO NUMBER:

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

```
int main(){
```

```
    int a,b,c;
```

```
    a=20;
```

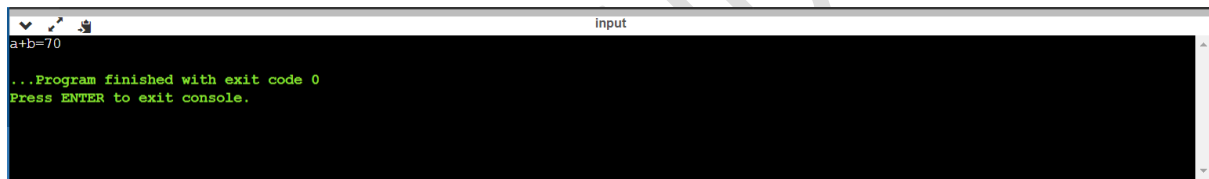
```
    b=50;
```

```
    c=a+b;
```

```
    printf("a+b=%d",c);
```

```
    return 0;
```

```
}
```



3) WAP TO FIND AREA OF CIRCLE:

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

```
int main(){
```

```
    float pie=3.14;
```

```
    int radius=10;
```

```
    printf("the radius of circle is %d",radius);
```

```
    float area=(float)(pie*radius*radius);
```

```
    printf("the area of circle is %f",area);
```

```
    return 0;
```

```
}
```

```
input
the radius of circle is 10the area of circle is 314.000000
...Program finished with exit code 0
Press ENTER to exit console.
```

4) WAP TO DIVIDE TWO NUMBER:

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

```
int main(){
```

```
    int a,b,c;
```

```
    a= 70;
```

```
    b= 20;
```

```
    c= a/b;
```

```
    printf("a/b=%d",c);
```

```
    return 0;
```

```
}
```

```
input
a/b=3
...Program finished with exit code 0
Press ENTER to exit console.
```

5) WAP TO PRINT ASCII VALUE:

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

```
int main(){
```

```
    char ch;
```

```
    printf("enter a character\n");
```

```
    scanf("%c",&ch);
```

```
    printf("ASCII value of %c is %d\n",ch,ch);
```

```
    return 0;
```

```
}
```



```
enter a character
A
ASCII value of A is 65

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

6) WAP TO MULTIPLY FLOATING POINT NUMBERS:

```
#include<stdio.h>

int main(){

    float num1,num2,result;

    printf("enter the first number:");

    scanf("%f",&num1);

    printf("enter the second number: ");

    scanf("%f",&num2);

    result=num1*num2;

    printf("the multiplication value is=%.2f",result);

    return 0;

}
```



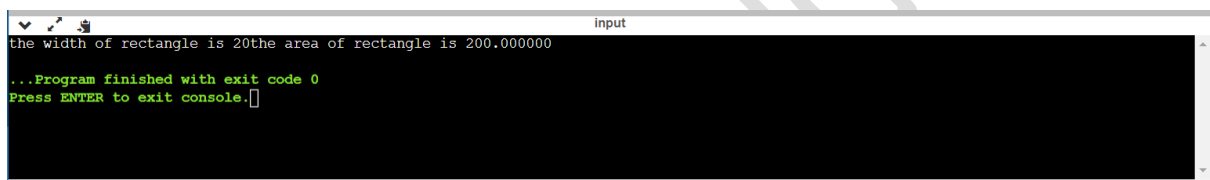
```
enter the first number:25.66
enter the second number: 45.78
the multiplication value is=1174.71

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

7) WAP TO FIND AREA OF RECTANGLE:

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

```
int main(){  
    int lenght=10;  
    int width=20;  
    printf("the width of rectangle is %d",width);  
    float area=(lenght*width);  
    printf("the area of rectangle is %f",area);  
    return 0;  
}
```



```
Input  
the width of rectangle is 20the area of rectangle is 200.000000  
...Program finished with exit code 0  
Press ENTER to exit console.
```

8) WAP TO FIND THE AREA OF SQUARE:

```
#include <stdio.h>

int main(){
    int side=12;
    printf("the side of square is %d",side);
    float area=(side*side);
    printf("the area pf square is %f",area);
    return 0;
}
```

A screenshot of a terminal window showing the execution of a C program. The output is: "the side of square is 12the area pf square is 144.000000". Below this, it says "...Program finished with exit code 0" and "Press ENTER to exit console." The terminal window has a title bar with "input" and standard window controls.

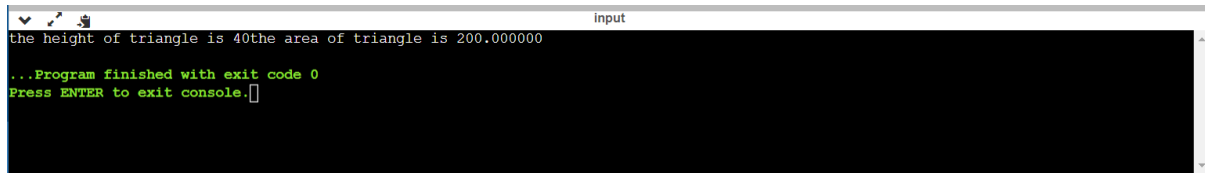
9) WAP TO FIND THE AREA OF RIGHT ANGLE TRIANGLE, ISOSCELES TRIANGLE, ANY TRIANGLE WITH THREE SIDES:

```
#include <stdio.h>

int main(){
    int base =10;
    int height=40;
    printf("the height of triangle is %d",height);
    float area=0.5*(base*height);
```



```
printf("the area of triangle is %f",area);  
  
return 0;  
  
}
```



```
input  
the height of triangle is 40the area of triangle is 200.000000  
...Program finished with exit code 0  
Press ENTER to exit console.
```

10) WAP TO FIND AREA AND VOLUME OF CUBE:

- **FOR AREA OF CUBE:**

```
#include <stdio.h>  
int main(){  
    int side =10;  
    printf("the side of cube is %d",side);  
    float area=(6*side*side);  
    printf("the area of cube is %f",area);  
    return 0;  
}
```



```
input  
the side of cube is 10the area of cube is 600.000000  
...Program finished with exit code 0  
Press ENTER to exit console.
```

- **FOR VOLUME OF CUBE:**

```
#include <stdio.h>  
int main(){  
    int side =15;
```

```

printf("the side of cube is %d",side);
float volume=(side*side*side);
printf("the volume of cube is %f",volume);
return 0;
}

```

```

the side of cube is 15the volume of cube is 3375.000000
...Program finished with exit code 0
Press ENTER to exit console.

```

11) WAP TO FIND THE AREA AND VOLUME OF CUBOID:

- **FOR AREA OF CUBOID:**

```

#include <stdio.h>
int main(){
    int lenght =40;
    int breadth=5;
    int height=6;
    printf("the lenght of cuboid is %d",lenght);
    float area =2*(lenght*breadth+breadth*height+lenght*height);
    printf("the area of cuboid is %f",area);
    return 0;
}

```

```

the lenght of cuboid is 40the area of cuboid is 940.000000
...Program finished with exit code 0
Press ENTER to exit console.

```

- **FOR VOLUME OF CUBOID:**

```

#include <stdio.h>
int main(){
    int lenght =40;

```

```

    int breadth=5;
    int height=6;
    printf("the lenght of cuboid is %d",lenght);
    float volume =(lenght*breadth*height);
    printf("the volume of cuboid is %f",volume);
    return 0;
}

```

```

input
the lenght of cuboid is 40the volume of cuboid is 1200.000000
...Program finished with exit code 0
Press ENTER to exit console.

```

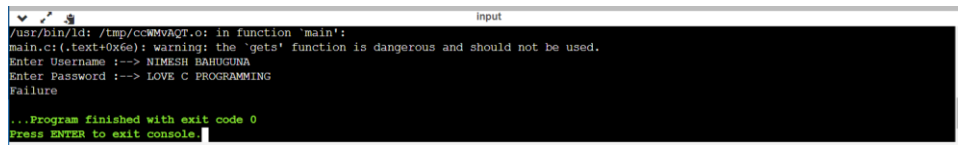
12) WAP TO VALIDATE THE USERID NAD PASWORD ENTERD BY USER IS CORRECCT OR NOT USING PREDEFINED USERNAME AND PASSWORD.

```

#include <stdio.h>
#include <string.h>
int main()
{
    int flag1 = 0,flag2 = 0;
    char user_name[] = "CodeCrucks";
    char user_password[] = "cc@123";
    char user_name1[20];
    char user_password1[20];
    printf("Enter Username :--> ");
    gets(user_name1);
    printf("Enter Password :--> ");
    gets(user_password1);
    if((strcmp(user_name, user_name1) == 0) && (strcmp(user_password, user_password1) == 0 ))
    {
        printf("Success");
    }
    else
    {
        printf("Failure");
    }
}

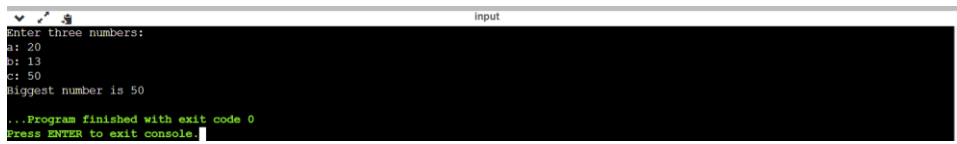
```

```
}  
return 0;  
}
```

A screenshot of a terminal window titled 'input'. The terminal shows the following text:
/usr/bin/ld: /tmp/ccNMVAGT.o: in function 'main':
main.c:(.text+0x6e): warning: the 'gets' function is dangerous and should not be used.
Enter Username :--> NIMESH BAHUGUNA
Enter Password :--> LOVE C PROGRAMMING
Failure
...Program finished with exit code 0
Press ENTER to exit console

13) WAP TO FIND THE LARGEST NUMBER USING LOGICAL AND OPERATOR.

```
#include <stdio.h>  
  
int main()  
{  
    int a, b, c;  
    printf("Enter three numbers: \na: ");  
    scanf("%d", &a);  
    printf("b: ");  
    scanf("%d", &b);  
    printf("c: ");  
    scanf("%d", &c);  
    if (a > b && a > c)  
        printf("Biggest number is %d", a);  
    if (b > a && b > c)  
        printf("Biggest number is %d", b);  
    if (c > a && c > b)  
        printf("Biggest number is %d", c);  
    return 0;  
}
```



```
Enter three numbers:
a: 20
b: 13
c: 50
Biggest number is 50
...Program finished with exit code 0
Press ENTER to exit console.
```

14. WAP TO INPUT THE POSITIVE NUMBER FROM THE USER TO PERFORM THE LEFT SHIFT OPERATOR.

```
#include <stdio.h>
int main ()
{
    // declare local variable
    int num;
    printf (" Enter a positive number: ");
    scanf ("%d", &num);
    // use left shift operator to shift the bits
    num = (num << 2); // It shifts two bits at the left side
    printf (" \n After shifting the binary bits to the left side. ");
    printf (" \n The new value of the variable num = %d", num);
    return 0;
}
```



```
Enter a positive number: 42
After shifting the binary bits to the left side.
The new value of the variable num = 168
...Program finished with exit code 0
Press ENTER to exit console.
```

15) WAP TO INPUT THE POSITIVE NUMBER FROM THE USER TO PERFORM THE RIGHT SHIFT.

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

```

{
// declare local variable
int num;
printf (" Enter a positive number: ");
scanf (" %d", &num);
// use right shift operator to shift the bits
num = (num >> 2); // It shifts two bits at the right side
printf (" \n After shifting the binary bits to the right side. ");
printf (" \n The new value of the variable num = %d", num);
return 0;
}

```

```

input
Enter a positive number: 2
After shifting the binary bits to the right side.
The new value of the variable num = 0
...Program finished with exit code 0
Press ENTER to exit console.

```

16) WAP TO PERFORM PRE INCREMENT AND PRE DECREMENT OPERATOR ON TWO INTEGERS AND PRINT BOTH ORIGINAL AND UPDATED VALUE.

PRE INCREMENT :-

```

#include <stdio.h>

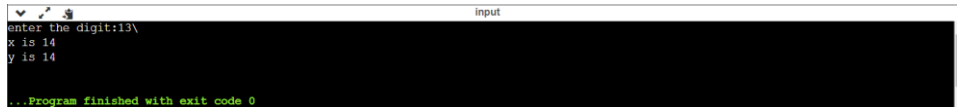
int main() {
    int x;
    printf("enter the digit:");
    scanf("%d",&x);

    int y = ++x;

    printf("x is %d\n", x);
    printf("y is %d\n", y);

    return 0;
}

```



```
input
enter the digit:13\
x is 14
y is 14
...Program finished with exit code 0
```

PRE DECREMENT:-

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

```
int main() {
```

```
    int x;
```

```
    printf("enter the digit:");
```

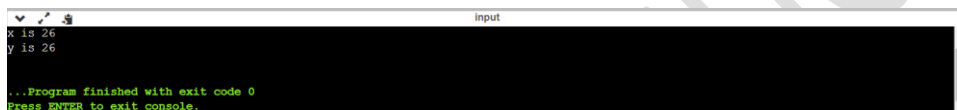
```
    scanf("%d",&x);
```

```
    int y = --x;
```

```
    printf("x is %d\n", x);
```

```
    printf("y is %d\n", y);
```

```
    return 0; }
```



```
input
x is 26
y is 26
...Program finished with exit code 0
Press ENTER to exit console.
```

17) WAP TO PERFORM POST INCREMENT AND POST DECREMENT OPERATOR ON TWO INTEGERS AND PRINT BOTH ORIGINAL AND UPDATED VALUE.

POST INCREMENT :-

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

```
int main() {
```

```
    int x;
```

```
    printf("enter the digit:");
```

```
    scanf("%d",&x);
```

```
    int y = x++;
```

```
    printf("x is %d\n", x);
```

```
    printf("y is %d\n", y);
```

```
    return 0;
```

```
}
```



A terminal window titled 'input' showing the following output: 'enter the digit:5', 'x is 6', 'y is 5', and '...Program finished with exit code 0'.

POST DECREMENT :-

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

```
int main() {
```

```
    int x;
```

```
    printf("enter the digit:");
```

```
    scanf("%d",&x);
```

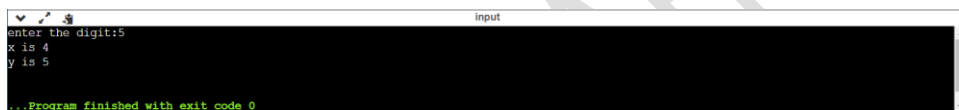
```
    int y = x--;
```

```
    printf("x is %d\n", x);
```

```
    printf("y is %d\n", y);
```

```
    return 0;
```

```
}
```



A terminal window titled 'input' showing the following output: 'enter the digit:5', 'x is 4', 'y is 5', and '...Program finished with exit code 0'.

18) WAP FOR AN INTEGER NUMBER AND TO CHECK WHETHER IT IS DIVISIBLE BY 9 OR 7 USING OR OPERATOR.

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

```
int main(){
```

```
    int x,y;
```

```
    printf("enter the number:");
```

```
    scanf("%d",&x);
```

```
    if((x% 9 == 0) || (x % 7 == 0)){
```

```
        printf("divisible");}
```

```
}
```



```
input
Enter the number:9
Divisible
...Program finished with exit code 0
Press ENTER to exit console.
```

19) WAP TO IDENTIFY GENDER IN SINGLE CHARACTER AND PRINT FULL GENDER (EX: IF INPUT IS M OR "m" -- IT SHOULD PRINT "MALE").

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

```
int main(){
```

```
    char gender;
```

```
    printf("Enter gender (M/m: ");
```

```
    scanf("%c",&gender);
```

```
    switch(gender){
```

```
        case 'M':
```

```
        case 'm':
```

```
            printf("Male.");
```

```
    }
```

```
    return 0;
```

```
}
```

```
input
Enter gender (M/m: M
Male.
...Program finished with exit code 0
Press ENTER to exit console.
```

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

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

```
int main() {
```

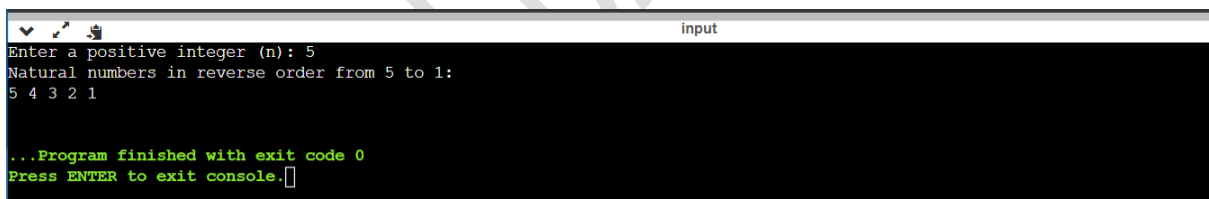
```

int n;

printf("Enter a positive integer (n): ")
scanf("%d", &n);
if (n <= 0) {
    printf("Please enter a positive integer.\n");
} else {
    printf("Natural numbers in reverse order from %d to 1:\n", n);
    for (int i = n; i >= 1; i--) {
        printf("%d ", i);
    }
    printf("\n");
}

return 0;
}

```



```

input
Enter a positive integer (n): 5
Natural numbers in reverse order from 5 to 1:
5 4 3 2 1

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

```

21) Write a C program to print all alphabets from a to z.

```

#include <stdio.h>

int main() {
    char alphabet;
    printf("Alphabets from 'a' to 'z':\n");
    for (alphabet = 'a'; alphabet <= 'z'; alphabet++) {
        printf("%c ", alphabet);
    }
}

```

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



```
Alphabets from 'a' to 'z':  
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.
```

22) Write a C program to print all natural numbers from 1 to n

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter a positive integer (n): ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 0) {
```

```
        printf("Please enter a positive integer.\n");
```

```
    } else {
```

```
        printf("Natural numbers in reverse order from %d to 1:\n", n);
```

```
        for (int i = 1; i <= n; i++) {
```

```
            printf("%d ", i);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```

```
input
Enter a positive integer (n): 5
Natural numbers in reverse order from 5 to 1:
1 2 3 4 5

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

23) write a program to print all even numbers between 1 to 100.

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

```
int main() {
```

```
    int n;
```

```
    if(n=100) {
```

```
        printf("Natural numbers in reverse order from %d to 1:\n", n);
```

```
        for (int i = 2; i < n; i++) {
```

```
            printf("%d ", i);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```

```
input
Natural numbers in reverse order from 100 to 1:
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94
95 96 97 98 99
```

24) Write a C program to print all odd number between 1 to 100

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

```

int main() {

    printf("Odd numbers between 1 and 100:\n");

    for (int i = 1; i <= 100; i++) {

        if (i % 2 != 0) {

            printf("%d ", i);

        }

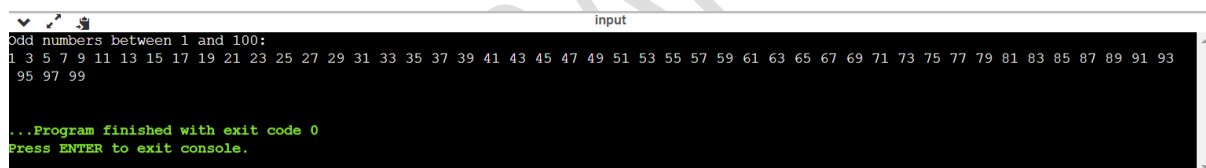
    }

    printf("\n");

    return 0;

}

```



```

input
Odd numbers between 1 and 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

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

```

25) Write a C program to find sum of all natural numbers between 1 to n

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

```

int main() {

    int n, sum = 0;

    printf("Enter a positive integer (n): ");

    scanf("%d", &n);

    if (n <= 0) {

        printf("Please enter a positive integer.\n");
    }
}

```

```

    } else {
        for (int i = 1; i <= n; i++) {
            sum += i;
        }
        printf("Sum of natural numbers from 1 to %d is: %d\n", n, sum);
    }

    return 0;
}

```

```

input
Enter a positive integer (n): 5
Sum of natural numbers from 1 to 5 is: 15

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

```

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

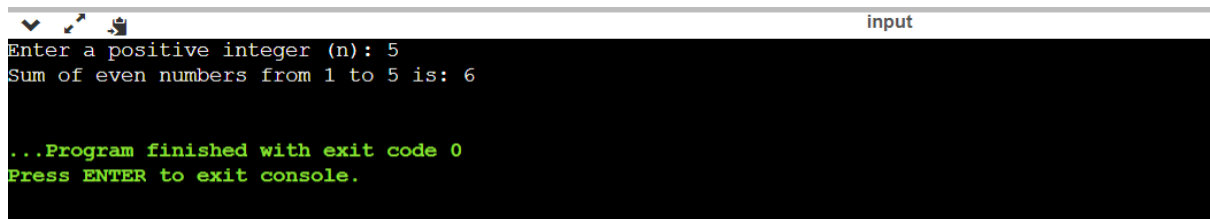
```
#include <stdio.h>
```

```

int main() {
    int n, sum = 0;
    printf("Enter a positive integer (n): ");
    scanf("%d", &n);
    if (n <= 0) {
        printf("Please enter a positive integer.\n");
    } else {
        for (int i = 2; i <= n; i += 2) {
            sum += i;
        }
        printf("Sum of even numbers from 1 to %d is: %d\n", n, sum);
    }
}

```

```
}  
  
    return 0;  
  
}
```

A screenshot of a console window with a dark background. The title bar at the top shows a standard Windows icon set and the text 'input'. The console text is as follows:
Enter a positive integer (n): 5
Sum of even numbers from 1 to 5 is: 6

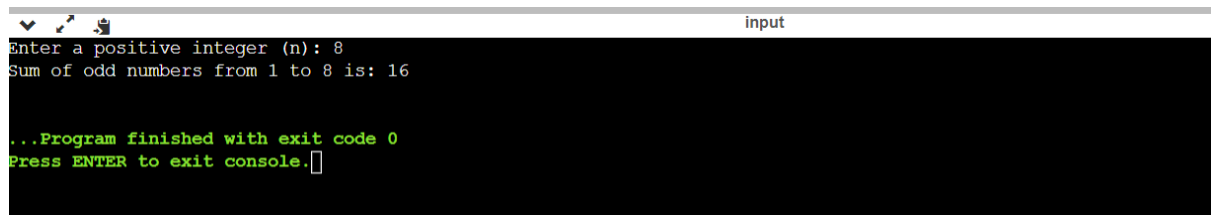
...Program finished with exit code 0
Press ENTER to exit console.
The text is displayed in a monospaced font, with the first two lines in white and the last two lines in green.

27) Write a C program to find sum of all odd numbers between 1 to n

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

```
int main() {  
    int n, sum = 0;  
    printf("Enter a positive integer (n): ");  
    scanf("%d", &n);  
    if (n <= 0) {  
        printf("Please enter a positive integer.\n");  
    } else {  
        for (int i = 1; i <= n; i += 2) {  
            sum += i;  
        }  
        printf("Sum of odd numbers from 1 to %d is: %d\n", n, sum);  
    }  
  
    return 0;
```

```
}
```



```
input
Enter a positive integer (n): 8
Sum of odd numbers from 1 to 8 is: 16

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

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

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

```
int main() {
    int number;

    printf("Enter a number for the multiplication table: ");
    scanf("%d", &number);

    printf("Multiplication table for %d:\n", number);
    for (int i = 1; i <= 10; i++) {
        printf("%d x %d = %d\n", number, i, number * i);
    }

    return 0;
}
```



```
input
Enter a number for the multiplication table: 4
Multiplication table for 4:
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
4 x 10 = 40

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


29) Write a C program to count number of digits in a number.

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

```
int main() {  
    int number, count = 0;  
    printf("Enter a number: ");  
    scanf("%d", &number);  
    int absoluteNumber = (number < 0) ? -number : number;  
    do {  
        count++;  
        absoluteNumber /= 10;  
    } while (absoluteNumber != 0);  
    printf("Number of digits in the entered number: %d\n", count);  
  
    return 0;  
}
```



```
input  
Enter a number: 45678  
Number of digits in the entered number: 5  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

30) 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);

int absoluteNumber = (number < 0) ? -number : number;

lastDigit = absoluteNumber % 10;

while (absoluteNumber >= 10) {

    absoluteNumber /= 10;

}

firstDigit = absoluteNumber;

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

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

return 0;

}

```

```

input
Enter a number: 45093
First digit: 4
Last digit: 3

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

```

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

```

#include <stdio.h>

int main() {

    int number, firstDigit, lastDigit, sum;

    printf("Enter a number: ");

    scanf("%d", &number);

    int absoluteNumber = (number < 0) ? -number : number;

    lastDigit = absoluteNumber % 10;

    while (absoluteNumber >= 10) {

        absoluteNumber /= 10;
    }
}

```

```

}

firstDigit = absoluteNumber;

sum = firstDigit + lastDigit;

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

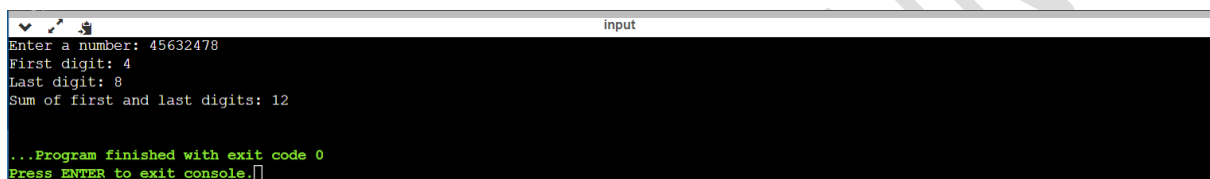
printf("Last digit: %d\n", lastDigit);

printf("Sum of first and last digits: %d\n", sum);

return 0;

}

```



```

input
Enter a number: 45632478
First digit: 4
Last digit: 8
Sum of first and last digits: 12

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

```

32) Write a C program to calculate sum of digits of a number

```

#include <stdio.h>

int main() {
    int number, digit, sum = 0;
    printf("Enter a number: ");
    scanf("%d", &number);
    int absoluteNumber = (number < 0) ? -number : number;
    while (absoluteNumber > 0) {
        digit = absoluteNumber % 10;
        sum += digit;
        absoluteNumber /= 10;
    }
    printf("Sum of the digits of the entered number: %d\n", sum);

    return 0;
}

```

```
}
```

```
input
Enter a number: 72649
Sum of the digits of the entered number: 28

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

33) Write a C program to calculate product of digits of a number

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

```
int main() {
    int number, digit, product = 1;
    printf("Enter a number: ");
    scanf("%d", &number);
    int absoluteNumber = (number < 0) ? -number : number;
    while (absoluteNumber > 0) {
        digit = absoluteNumber % 10;
        product *= digit;
        absoluteNumber /= 10;
    }
    printf("Product of the digits of the entered number: %d\n", product);

    return 0;
}
```


```
input
Enter a number: 34986
Product of the digits of the entered number: 5184

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

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

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

```
int main() {  
    int number, reversedNumber = 0, remainder;  
    printf("Enter a number: ");  
    scanf("%d", &number);  
    while (number != 0) {  
        remainder = number % 10;  
        reversedNumber = reversedNumber * 10 + remainder;  
        number /= 10;  
    }  
    printf("Reversed number: %d\n", reversedNumber);  
  
    return 0;  
}
```

A screenshot of a terminal window titled 'input'. The terminal shows the execution of a C program. It prompts 'Enter a number: 746' and outputs 'Reversed number: 647'. Below this, it shows '...Program finished with exit code 0' and 'Press ENTER to exit console.' with a cursor. The terminal has a dark background with light-colored text.

```
input  
Enter a number: 746  
Reversed number: 647  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

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

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

```
int main() {  
    int number, originalNumber, reversedNumber = 0, remainder;
```

```

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

originalNumber = number;
while (number != 0) {
    remainder = number % 10;
    reversedNumber = reversedNumber * 10 + remainder;
    number /= 10;
}
if (originalNumber == reversedNumber) {
    printf("%d is a palindrome number.\n", originalNumber);
} else {
    printf("%d is not a palindrome number.\n", originalNumber);
}

return 0;
}

```



```

input
Enter a number: 2222222
2222222 is a palindrome number.

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

```

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

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

```

int main() {
    int number, digit, count[10] = {0};
    printf("Enter an integer: ");
    scanf("%d", &number);

```

```

int absoluteNumber = (number < 0) ? -number : number;

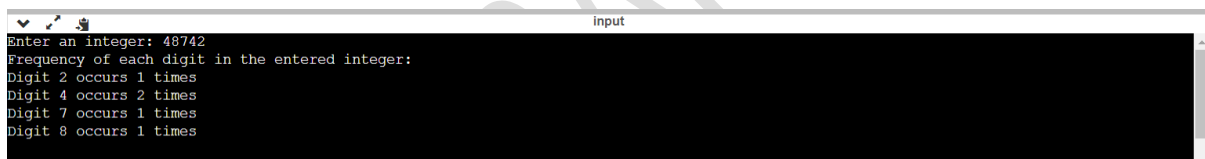
while (absoluteNumber > 0) {
    digit = absoluteNumber % 10;
    count[digit]++;
    absoluteNumber /= 10;
}

printf("Frequency of each digit in the entered integer:\n");

for (int i = 0; i < 10; i++) {
    if (count[i] > 0) {
        printf("Digit %d occurs %d times\n", i, count[i]);
    }
}

return 0;
}

```



```

Enter an integer: 48742
Frequency of each digit in the entered integer:
Digit 2 occurs 1 times
Digit 4 occurs 2 times
Digit 7 occurs 1 times
Digit 8 occurs 1 times

```

37) Write a C program to enter a number and print it in words

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

```
void numberToWords(int num) {
```

```
    char *ones[] = {"", "One", "Two", "Three", "Four", "Five", "Six", "Seven", "Eight", "Nine"};
```

```
    char *tens[] = {"", "Ten", "Twenty", "Thirty", "Forty", "Fifty", "Sixty", "Seventy", "Eighty", "Ninety"};
```

```
    char *teens[] = {"", "Eleven", "Twelve", "Thirteen", "Fourteen", "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen"};
```

```
    if (num < 1 || num > 99) {
```

```
        printf("Number out of range (1-99)\n");
```

```

        return;
    }
    if (num >= 11 && num <= 19) {
        printf("%s\n", teens[num - 10]);
    } else {
        int tensDigit = num / 10;
        int onesDigit = num % 10;

        printf("%s%s\n", tens[tensDigit], ones[onesDigit]);
    }
}

int main() {
    int num;
    printf("Enter a number (1-99): ");
    scanf("%d", &num);
    numberToWords(num);

    return 0;
}

```



```

input
Enter a number (1-99): 77
SeventySeven

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

```

38) Write a C program to print all ASCII character with their values

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

```
int main() {
```



```

printf("ASCII Characters and Their Values (0-127):\n");

for (int i = 0; i <= 127; i++) {
    printf("ASCII Value %d: %c\n", i, (char)i);
}

return 0;
}

```



```

ASCII Value 33: !
ASCII Value 34: "
ASCII Value 35: #
ASCII Value 36: $
ASCII Value 37: %
ASCII Value 38: &
ASCII Value 39: '
ASCII Value 40: (
ASCII Value 41: )
ASCII Value 42: *
ASCII Value 43: +
ASCII Value 44: ,
ASCII Value 45: -
ASCII Value 46: .
ASCII Value 47: /
ASCII Value 48: 0
ASCII Value 49: 1
ASCII Value 50: 2
ASCII Value 51: 3
ASCII Value 52: 4
ASCII Value 53: 5
ASCII Value 54: 6
ASCII Value 55: 7
ASCII Value 56: 8
ASCII Value 57: 9
ASCII Value 58: :
ASCII Value 59: ;
ASCII Value 60: <
ASCII Value 61: =
ASCII Value 62: >

```

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

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

```

double power(double base, int exponent) {
    double result = 1.0;

    for (int i = 0; i < exponent; i++) {
        result *= base;
    }
}

```

```

    return result;
}

int main() {
    double base, result;
    int exponent;
    printf("Enter the base: ");
    scanf("%lf", &base);
    printf("Enter the exponent: ");
    scanf("%d", &exponent);
    result = power(base, exponent);
    printf("%.2lf ^ %d = %.2lf\n", base, exponent, result);

    return 0;
}

```

```

input
Enter the base: 10
Enter the exponent: 5
10.00 ^ 5 = 100000.00

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

```

40) Write a C program to find all factors of a number

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

```

int main() {
    int number;

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

    printf("Factors of %d are: ", number);
    for (int i = 1; i <= number; i++) {

```

```

        if (number % i == 0) {
            printf("%d ", i);
        }
    }
    printf("\n");

    return 0;
}

```

```

input
Enter a number: 20
Factors of 20 are: 1 2 4 5 10 20

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

```

41) Write a C program to calculate factorial of a number

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

```

int main() {
    int n;
    unsigned long long factorial = 1;
    printf("Enter a non-negative integer: ");
    scanf("%d", &n);
    if (n < 0) {
        printf("Factorial is not defined for negative numbers.\n");
    } else {
        for (int i = 1; i <= n; i++) {
            factorial *= i;
        }
        printf("Factorial of %d is %llu\n", n, factorial);
    }
}

```

```
    return 0;
}
```

A screenshot of a console window with a black background and white text. The window title bar shows a standard icon and the word 'input'. The text inside the console reads: 'Enter a non-negative integer: 40', 'Factorial of 40 is 18376134811363311616', '...Program finished with exit code 0', and 'Press ENTER to exit console.'

```
input
Enter a non-negative integer: 40
Factorial of 40 is 18376134811363311616
...Program finished with exit code 0
Press ENTER to exit console.
```

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

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

```
int findHCF(int a, int b) {
    while (a != b) {
        if (a > b) {
            a = a - b;
        } else {
            b = b - a;
        }
    }
    return a;
}
```

```
int main() {
    int num1, num2;

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

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

    int hcf = findHCF(num1, num2);

    printf("HCF (GCD) of %d and %d is %d\n", num1, num2, hcf);
}
```

```
    return 0;
}
```



```
input
Enter the first number: 4
Enter the second number: 8
HCF (GCD) of 4 and 8 is 4

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

43) Write a C program to find LCM of two numbers

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

```
int findHCF(int a, int b) {
    while (a != b) {
        if (a > b) {
            a = a - b;
        } else {
            b = b - a;
        }
    }
    return a;
}
```

```
int findLCM(int a, int b) {
    int hcf = findHCF(a, b);
    int lcm = (a * b) / hcf;
    return lcm;
}
```

```
int main() {
    int num1, num2;
```

```

printf("Enter the first number: ");

scanf("%d", &num1);

printf("Enter the second number: ");

scanf("%d", &num2);

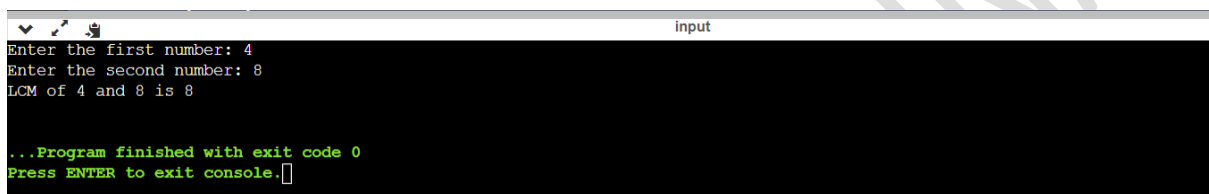
int lcm = findLCM(num1, num2);

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


return 0;

}

```



```

input
Enter the first number: 4
Enter the second number: 8
LCM of 4 and 8 is 8

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

```

44) Write a C program to check whether a number is Prime number or not

```

#include <stdio.h>

int main(){

    int n,i,count=0;

    printf ("enter any number: ");

    scanf ("%d",&n);

    for(i=1;i<=n;i++){

        if (n%i==1){

            count++;

        }

    }if (count==2)

    printf ("prime number");

    else

    printf ("non prime");

    return 0;
}

```

```
}
```



```
input
enter any number: 8
non prime
...Program finished with exit code 0
Press ENTER to exit console.
```

45) 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;
```

```
    }
```

```
    for (int i = 2; i * i <= num; i++) {
```

```
        if (num % i == 0) {
```

```
            return 0;
```

```
        }
```

```
    }
```

```
    return 1;
```

```
}
```

```
int main() {
```

```
    int n;
```

```
    printf("Enter a positive integer (n): ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 1) {
```

```
        printf("There are no prime numbers in the range 1 to %d\n", n);
```

```

    } else {

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

        for (int i = 2; i <= n; i++) {

            if (isPrime(i)) {

                printf("%d ", i);

            }

        }

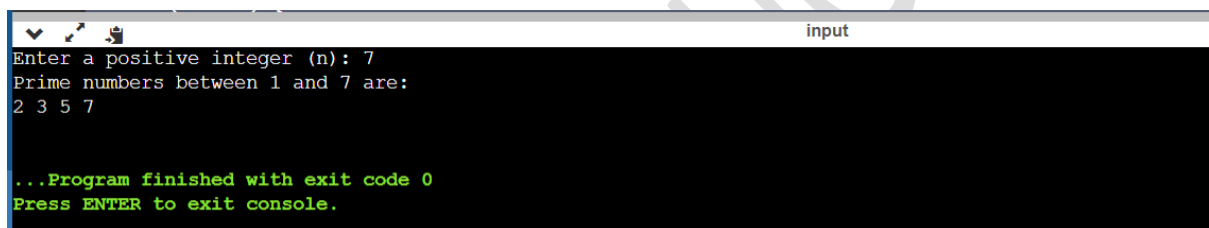
        printf("\n");

    }

    return 0;

}

```



```

input
Enter a positive integer (n): 7
Prime numbers between 1 and 7 are:
2 3 5 7

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

```

46) 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;
- }
- for (int i = 2; i * i <= num; i++) {
- if (num % i == 0) {
- return 0;
- }
- }
- return 1;


```

• }

• int main() {
•     int n, sum = 0;
•     printf("Enter a positive integer (n): ");
•     scanf("%d", &n);

•     if (n <= 1) {
•         printf("There are no prime numbers in the range 1 to %d\n", n);
•     } else {
•         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("\n");
•         printf("Sum of prime numbers between 1 and %d is: %d\n", n, sum);
•     }

•     return 0;
}

```



```

input
Enter a positive integer (n): 9
Prime numbers between 1 and 9 are:
2 3 5 7
Sum of prime numbers between 1 and 9 is: 17

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

```

47) Write a C program to find all prime factors of a number

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

```

int isPrime(int num) {
    if (num <= 1) {
        return 0;
    }
}

```

```
}  
for (int i = 2; i * i <= num; i++) {  
    if (num % i == 0) {  
        return 0;  
    }  
}  
return 1;  
}
```

```
int main() {  
    int number;  
    printf("Enter a positive integer: ");  
    scanf("%d", &number);  
  
    printf("Prime factors of %d are: ", number);  
  
    if (number <= 1) {  
        printf("There are no prime factors for %d.\n", number);  
    } else {  
        for (int i = 2; i <= number; i++) {  
            if (number % i == 0 && isPrime(i)) {  
                while (number % i == 0) {  
                    printf("%d ", i);  
                    number = number / i;  
                }  
            }  
        }  
  
        printf("\n");  
    }  
  
    return 0;
```

```
}
```

```
input
Enter a positive integer: 4'
Prime factors of 4 are: 2 2

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

48) 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, 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 num;
```

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

```
    scanf("%d", &num);
```

```
    if (isArmstrong(num))
```

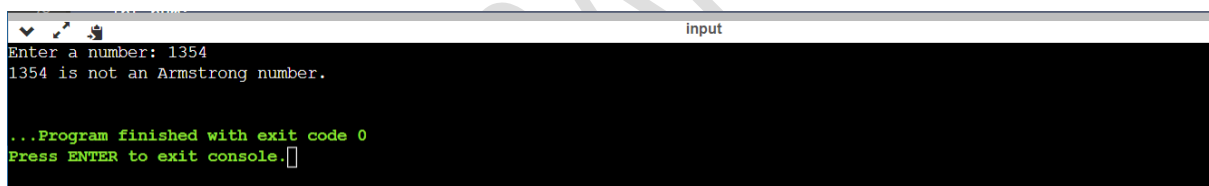
```
        printf("%d is an Armstrong number.\n", num);
```

```
    else
```

```
        printf("%d is not an Armstrong number.\n", num);
```

```
    return 0;
```

```
}
```



```
Enter a number: 1354
1354 is not an Armstrong number.

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

49) 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;
}
```

```
}
```

```
input
Enter the value of n: 5
Armstrong numbers between 1 and 5 are:
1
2
3
4
5
```

50) Write a C program to check whether a number is Perfect number or not

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

```
int isPerfect(int num) {
    int sum = 0;
    for (int i = 1; i <= num / 2; i++) {
        if (num % i == 0) {
            sum += i;
        }
    }
    if (sum == num) {
        return 1;
    } else {
        return 0;
    }
}
```

```
int main() {
    int num;

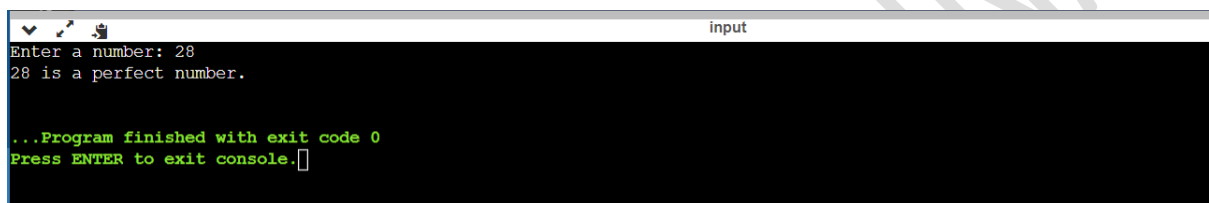
    printf("Enter a number: ");
    scanf("%d", &num);
```

```

if (isPerfect(num)) {
    printf("%d is a perfect number.\n", num);
} else {
    printf("%d is not a perfect number.\n", num);
}

return 0;
}

```



```

input
Enter a number: 28
28 is a perfect number.

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

```

51) 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;
        }
    }
    if (sum == num) {
        return 1;
    } else {
        return 0;
    }
}

```

```

    }
}

int main() {
    int n;

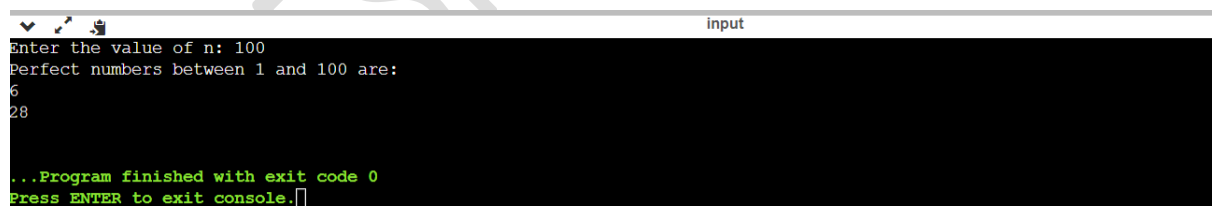
    printf("Enter the value of n: ");
    scanf("%d", &n);

    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;
}

```



```

input
Enter the value of n: 100
Perfect numbers between 1 and 100 are:
6
28
...Program finished with exit code 0
Press ENTER to exit console.

```

52) Write a C program to check whether a number is Strong number or not

```

#include <stdio.h>

int factorial(int num) {
    if (num == 0 || num == 1) {

```



```
        return 1;
    }
    return num * factorial(num - 1);
}
```

```
int isStrong(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 num;

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

    if (isStrong(num)) {
        printf("%d is a strong number.\n", num);
    } else {
        printf("%d is not a strong number.\n", num);
    }

    return 0;
}
```

```
}
```

```
input
Enter a number: 145
145 is a strong number.

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

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

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

```
int factorial(int num) {
    if (num == 0 || num == 1) {
        return 1;
    }
    return num * factorial(num - 1);
}
```

```
int isStrong(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 the value of n: ");


    scanf("%d", &n);

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

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

    return 0;
}

```



```

input
Enter the value of n: 100
Strong numbers between 1 and 100 are:
1
2
...Program finished with exit code 0
Press ENTER to exit console.

```

54) Write a C program to print Fibonacci series up to n terms

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

```

int main() {

    int n, first = 0, second = 1, next, i;

    printf("Enter the number of terms: ");

    scanf("%d", &n);

```

```

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

for (i = 0; i < n; i++) {
    if (i <= 1) {
        next = i;
    } else {
        next = first + second;
        first = second;
        second = next;
    }
    printf("%d", next);

    if (i < n - 1) {
        printf(", ");
    } else {
        printf("\n");
    }
}

return 0;
}

```



The screenshot shows a terminal window with a title bar that includes a window icon, a maximize icon, and a close icon, followed by the word "input". The terminal content is as follows:

```

Enter the number of terms: 9
Fibonacci Series up to 9 terms:
0, 1, 1, 2, 3, 5, 8, 13, 21

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

```

55) Write a C program to find one's complement of a binary number

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

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

```
void onesComplement(char binary[]) {
```

```
    int length = strlen(binary);
```

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

```
        if (binary[i] == '0') {
```

```
            binary[i] = '1';
```

```
        } else if (binary[i] == '1') {
```

```
            binary[i] = '0';
```

```
        }
```

```
    }
```

```
}
```

```
int main() {
```

```
    char binary[100];
```

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

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

```
    onesComplement(binary);
```

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

```
    return 0;
```

```
}
```

```
input
Enter a binary number: 1010001011
One's complement: 0101110100

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

56) Write a C program to find two's complement of a binary number

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

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

```
void onesComplement(char binary[]) {
```

```
    int length = strlen(binary);
```

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

```
        if (binary[i] == '0') {
```

```
            binary[i] = '1';
```

```
        } else if (binary[i] == '1') {
```

```
            binary[i] = '0';
```

```
        }
```

```
    }
```

```
}
```

```
void twosComplement(char binary[]) {
```

```
    onesComplement(binary);
```

```
    int length = strlen(binary);
```

```
    int carry = 1;
```

```
    for (int i = length - 1; i >= 0; i--) {
```

```
        if (binary[i] == '0' && carry == 1) {
```

```
            binary[i] = '1';
```

```
            carry = 0;
```

```
        } else if (binary[i] == '1' && carry == 1) {
```

```

        binary[i] = '0';
        carry = 1;
    }
}
}

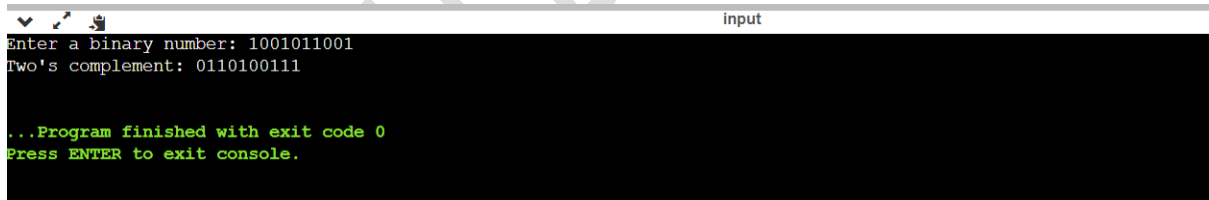
int main() {
    char binary[100];
    printf("Enter a binary number: ");
    scanf("%s", binary);

    twosComplement(binary);

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

    return 0;
}

```



```

input
Enter a binary number: 1001011001
Two's complement: 0110100111

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

```

57) Write a C program to convert Binary to Octal number system

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

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

```
// Function to convert a binary digit to octal
```

```
int binaryToOctal(char binaryDigit[3]) {
```

```

int octalDigit = 0;

// Convert binaryDigit to decimal
for (int i = 0; i < 3; i++) {
    octalDigit = octalDigit * 2 + (binaryDigit[i] - '0');
}

return octalDigit;
}

int main() {
    char binary[100];
    char binaryDigit[3];
    int length, i, j, k;

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

    length = strlen(binary);

    // Ensure that the binary number is divisible by 3, add leading zeros if necessary
    if (length % 3 != 0) {
        int newLength = (length + 2) / 3 * 3; // Round up to the nearest multiple of 3
        for (i = length; i < newLength; i++) {
            binary[i] = '0';
        }
        binary[newLength] = '\0';
        length = newLength;
    }

    printf("Octal representation: ");

```



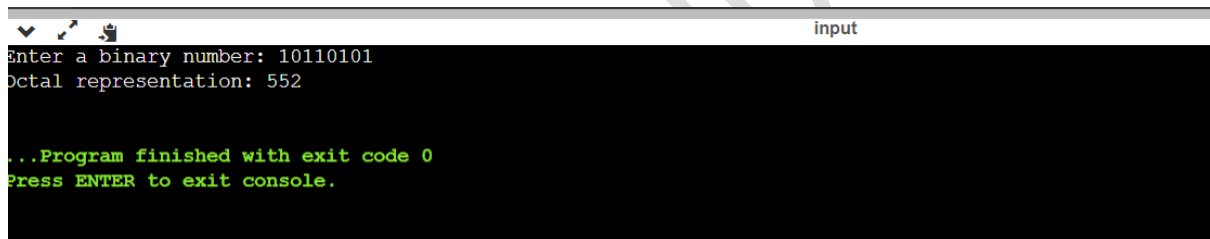
```

for (i = 0; i < length; i += 3) {
    for (j = i, k = 0; j < i + 3; j++, k++) {
        binaryDigit[k] = binary[j];
    }
    binaryDigit[3] = '\0';
    int octalDigit = binaryToOctal(binaryDigit);
    printf("%d", octalDigit);
}

printf("\n");

return 0;
}

```



```

input
Enter a binary number: 10110101
Octal representation: 552

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

```

58) Write a C program to convert Binary to Decimal number system

```

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

int binaryToDecimal(char binary[]) {
    int decimal = 0;
    int length = strlen(binary);

    for (int i = length - 1; i >= 0; i--) {
        if (binary[i] == '1') {

```

```

        decimal += 1 << (length - 1 - i);
    }
}

return decimal;
}

int main() {
    char binary[100];

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

    int decimal = binaryToDecimal(binary);

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

    return 0;
}

```



```

input
Enter a binary number: 10010110
Decimal representation: 150

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

```

59) Write a C program to convert Binary to Hexadecimal number system

```

#include <stdio.h>

#include <string.h>

char binaryToHex(char binary[4]) {
    int decimal = 0;

```

```

for (int i = 3; i >= 0; i--) {
    decimal += (binary[i] - '0') << (3 - i);
}

if (decimal >= 0 && decimal <= 9) {
    return (char)(decimal + '0');
} else {
    return (char)(decimal - 10 + 'A');
}
}

int main() {
    char binary[100];
    char binaryDigit[4];
    char hexadecimal[100];
    int length, i, j, k;

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

    length = strlen(binary);
    if (length % 4 != 0) {
        int newLength = (length + 3) / 4 * 4;
        for (i = length; i < newLength; i++) {
            binary[i] = '0';
        }
        binary[newLength] = '\0';
        length = newLength;
    }

    printf("Hexadecimal representation: 0x");

```

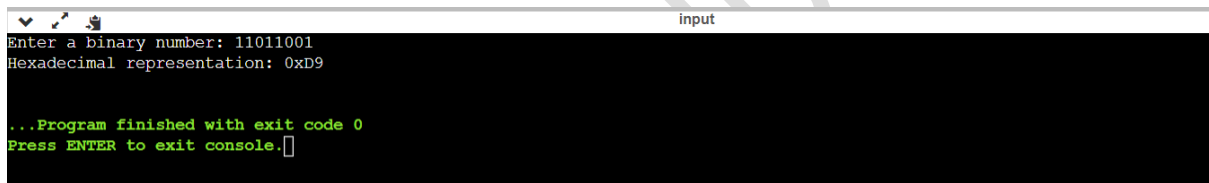
```

for (i = 0; i < length; i += 4) {
    for (j = i, k = 0; j < i + 4; j++, k++) {
        binaryDigit[k] = binary[j];
    }
    binaryDigit[4] = '\0';
    char hexDigit = binaryToHex(binaryDigit);
    printf("%c", hexDigit);
}

printf("\n");

return 0;
}

```



```

input
Enter a binary number: 11011001
Hexadecimal representation: 0xD9

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

```

60) Write a C program to convert Octal to Binary number system.

```

#include <stdio.h>
#include <string.h>
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 NULL;
    }
}
```

```
int main() {
    char octal[100];
    char binary[400];
    int length, i, j;

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

    length = strlen(octal);
    for (i = 0, j = 0; i < length; i++) {
        char* binaryDigit = octalToBinary(octal[i]);
        if (binaryDigit == NULL) {
            printf("Invalid octal digit: %c\n", octal[i]);
            return 1;
        }
        strcpy(binary + j, binaryDigit);
        j += 3;
    }
    binary[j] = '\0';

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

    return 0;
}
```

```
input
Enter an octal number: 56
Binary representation: 101110

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

61) Write a C program to convert Octal to Decimal number system

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

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

```
int octalToDecimal(char octal[]) {
```

```
    int decimal = 0;
```

```
    int length = strlen(octal);
```

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

```
        int octalDigit = octal[i] - '0';
```

```
        decimal = decimal * 8 + octalDigit;
```

```
    }
```

```
    return decimal;
```

```
}
```

```
int main() {
```

```
    char octal[100];
```

```
    printf("Enter an octal number: ");
```

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

```
    int decimal = octalToDecimal(octal);
```

```

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

return 0;
}

```



```

input
Enter an octal number: 45
Decimal representation: 37

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

```

62) Write a C program to convert Octal to Hexadecimal number system

```

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

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 NULL;
    }
}

char binaryToHex(char binary[4]) {
    int decimal = 0;
    for (int i = 3; i >= 0; i--) {

```

```

        decimal += (binary[i] - '0') << (3 - i);
    }

    if (decimal >= 0 && decimal <= 9) {
        return (char)(decimal + '0');
    } else {
        return (char)(decimal - 10 + 'A');
    }
}

int main() {
    char octal[100];
    char binary[400];
    char hex[400];
    int length, i, j;

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

    length = strlen(octal);
    for (i = 0, j = 0; i < length; i++) {
        char* binaryDigit = octalToBinary(octal[i]);
        if (binaryDigit == NULL) {
            printf("Invalid octal digit: %c\n", octal[i]);
            return 1;
        }
        strcpy(binary + j, binaryDigit);
        j += 3;
    }
    binary[j] = '\0';
    if (j % 4 != 0) {

```



```

int newLength = (j + 3) / 4 * 4;

for (i = j; i < newLength; i++) {
    binary[i] = '0';
}

binary[newLength] = '\0';

j = newLength;
}

printf("Hexadecimal representation: 0x");

for (i = 0; i < j; i += 4) {
    char binaryDigit[5];
    for (int k = 0; k < 4; k++) {
        binaryDigit[k] = binary[i + k];
    }
    binaryDigit[4] = '\0';
    char hexDigit = binaryToHex(binaryDigit);
    printf("%c", hexDigit);
}

printf("\n");

return 0;
}

```



```

input
Enter an octal number: 45
Hexadecimal representation: 0x94

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

```

63) Write a C program to convert Decimal to Binary number system

```
#include <stdio.h>

void decimalToBinary(int decimal) {
    int binary[32];
    int index = 0;
    if (decimal == 0) {
        printf("Binary representation: 0\n");
        return;
    }
    while (decimal > 0) {
        binary[index] = decimal % 2;
        decimal = decimal / 2;
        index++;
    }

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

int main() {
    int decimal;

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

    decimalToBinary(decimal);

    return 0;
}
```

```
input
Enter a decimal number: 56
Binary representation: 111000

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

64) Write a C program to convert Decimal to Octal number system

```
#include <stdio.h>

void decimalToOctal(int decimal) {
    int octal[32];
    int index = 0;
    if (decimal == 0) {
        printf("Octal representation: 0\n");
        return;
    }
    while (decimal > 0) {
        octal[index] = decimal % 8;
        decimal = decimal / 8;
        index++;
    }

    printf("Octal representation: ");
    for (int i = index - 1; i >= 0; i--) {
        printf("%d", octal[i]);
    }
    printf("\n");
}

int main() {
    int decimal;
```

```

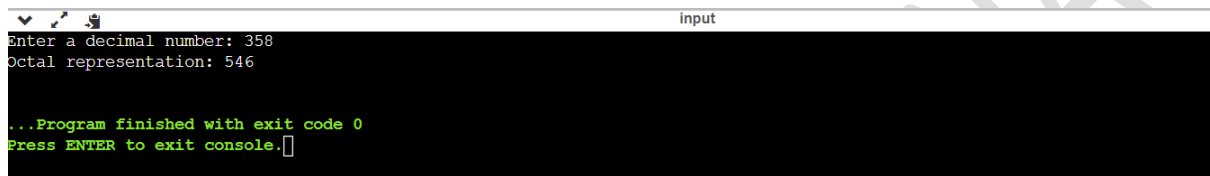
printf("Enter a decimal number: ");

scanf("%d", &decimal);

decimalToOctal(decimal);

return 0;
}

```



```

input
Enter a decimal number: 358
Octal representation: 546

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

```

65) Write a C program to convert Decimal to Hexadecimal number system

```

#include <stdio.h>

char decimalToHexDigit(int decimalDigit) {
    if (decimalDigit >= 0 && decimalDigit <= 9) {
        return (char)('0' + decimalDigit);
    } else {
        return (char)('A' + (decimalDigit - 10));
    }
}

void decimalToHexadecimal(int decimal) {
    char hexadecimal[32];
    int index = 0;
    if (decimal == 0) {
        printf("Hexadecimal representation: 0\n");
        return;
    }
    while (decimal > 0) {

```

```

    int remainder = decimal % 16;

    hexadecimal[index] = decimalToHexDigit(remainder);

    decimal = decimal / 16;

    index++;
}

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

printf("\n");
}

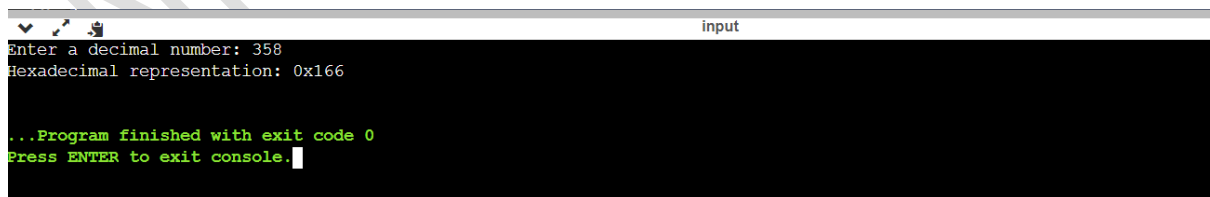
int main() {
    int decimal;

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

    decimalToHexadecimal(decimal);

    return 0;
}

```



```

input
Enter a decimal number: 358
Hexadecimal representation: 0x166

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

```

66) Write a C program to convert Hexadecimal to Binary number system.

```
#include <stdio.h>

#include <string.h>

char* hexToBinary(char hexDigit) {
    switch (hexDigit) {
        case '0': return "0000";
        case '1': return "0001";
        case '2': return "0010";
        case '3': return "0011";
        case '4': return "0100";
        case '5': return "0101";
        case '6': return "0110";
        case '7': return "0111";
        case '8': return "1000";
        case '9': return "1001";
        case 'A': case 'a': return "1010";
        case 'B': case 'b': return "1011";
        case 'C': case 'c': return "1100";
        case 'D': case 'd': return "1101";
        case 'E': case 'e': return "1110";
        case 'F': case 'f': return "1111";
        default: return NULL;
    }
}

int main() {
    char hexadecimal[100];
    char binary[400];
    int length, i, j;

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


```

length = strlen(hexadecimal);

printf("Binary representation: ");
for (i = 0, j = 0; i < length; i++) {
    char* binaryDigit = hexToBinary(hexadecimal[i]);
    if (binaryDigit == NULL) {
        printf("Invalid hexadecimal digit: %c\n", hexadecimal[i]);
        return 1;
    }
    printf("%s", binaryDigit);
}
printf("\n");

return 0;
}

```



```

input
Enter a hexadecimal number: 56A
Binary representation: 010101101010

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

```

67) Write a C program to convert Hexadecimal to Octal number system

```

#include <stdio.h>

#include <string.h>

char* hexToBinary(char hexDigit) {
    switch (hexDigit) {
        case '0': return "0000";
        case '1': return "0001";
        case '2': return "0010";

```

```

    case '3': return "0011";
    case '4': return "0100";
    case '5': return "0101";
    case '6': return "0110";
    case '7': return "0111";
    case '8': return "1000";
    case '9': return "1001";
    case 'A': case 'a': return "1010";
    case 'B': case 'b': return "1011";
    case 'C': case 'c': return "1100";
    case 'D': case 'd': return "1101";
    case 'E': case 'e': return "1110";
    case 'F': case 'f': return "1111";
    default: return NULL;
}
}

char binaryToOctal(char binary[3]) {
    int decimal = 0;
    for (int i = 2; i >= 0; i--) {
        decimal += (binary[i] - '0') << (2 - i);
    }
    return (char)('0' + decimal);
}

int main() {
    char hexadecimal[100];
    char binary[400];
    char octal[400];
    int length, i, j, k;

    printf("Enter a hexadecimal number: ");

```



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

```
length = strlen(hexadecimal);
```

```
printf("Octal representation: ");
```

```
for (i = 0, j = 0; i < length; i++) {
```

```
    char* binaryDigit = hexToBinary(hexadecimal[i]);
```

```
    if (binaryDigit == NULL) {
```

```
        printf("Invalid hexadecimal digit: %c\n", hexadecimal[i]);
```

```
        return 1;
```

```
    }
```

```
    strcpy(binary + j, binaryDigit);
```

```
    j += 4;
```

```
}
```

```
if (j % 3 != 0) {
```

```
    int newLength = (j + 2) / 3 * 3;
```

```
    for (k = j; k < newLength; k++) {
```

```
        binary[k] = '0';
```

```
    }
```

```
    binary[newLength] = '\0';
```

```
    j = newLength;
```

```
}
```

```
for (i = 0; i < j; i += 3) {
```

```
    char binaryDigit[4];
```

```
    for (k = 0; k < 3; k++) {
```

```
        binaryDigit[k] = binary[i + k];
```

```
    }
```

```
    binaryDigit[3] = '\0';
```

```
    char octalDigit = binaryToOctal(binaryDigit);
```

```
    printf("%c", octalDigit);
```

```

}

printf("\n");

return 0;

}

```



```

input
Enter a hexadecimal number: 1d7a
Octal representation: 072750

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

```

68) Write a C program to convert Hexadecimal to Decimal number system

```

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

int hexToDecimalDigit(char hexDigit) {
    if (hexDigit >= '0' && hexDigit <= '9') {
        return hexDigit - '0';
    } else if (hexDigit >= 'A' && hexDigit <= 'F') {
        return hexDigit - 'A' + 10;
    } else if (hexDigit >= 'a' && hexDigit <= 'f') {
        return hexDigit - 'a' + 10;
    } else {
        return -1;
    }
}

int hexToDecimal(char hexadecimal[]) {
    int decimal = 0;
    int length = strlen(hexadecimal);
    for (int i = 0; i < length; i++) {
        int decimalDigit = hexToDecimalDigit(hexadecimal[i]);

```

```

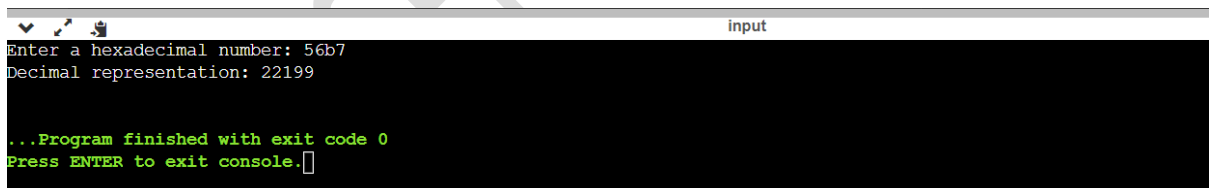
    if (decimalDigit == -1) {
        printf("Invalid hexadecimal digit: %c\n", hexadecimal[i]);
        return -1;
    }

    decimal = decimal * 16 + decimalDigit;
}

return decimal;
}

int main() {
    char hexadecimal[100];
    printf("Enter a hexadecimal number: ");
    scanf("%s", hexadecimal);
    int decimal = hexToDecimal(hexadecimal);
    if (decimal != -1) {
        printf("Decimal representation: %d\n", decimal);
    }
    return 0;
}

```



```

input
Enter a hexadecimal number: 56b7
Decimal representation: 22199

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

```

Pattern Exercises

69) pyramid star pattern

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

```

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

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

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

        printf("\n");
    }

    return 0;
}

```



```

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

```

70) hollow pyramid star pattern

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

```

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

```

```

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

for (i = 1; i <= rows; i++) {
    for (space = 1; space <= rows - i; space++) {
        printf(" ");
    }
    if (i == 1 || i == rows) {
        for (j = 1; j <= 2 * i - 1; j++) {
            printf("*");
        }
    } else {
        printf("*");
        for (j = 1; j <= 2 * i - 3; j++) {
            printf(" ");
        }
        printf("*");
    }

    printf("\n");
}

return 0;
}

```

The screenshot shows a terminal window with the following output:

```

Enter the number of rows: 7
  *
 * *
*   *
* * *
* * *
* * *
*****

```

The pattern is a reverse pyramid where the number of stars decreases from 7 in the first row to 1 in the seventh row. The stars are aligned to the left, and the rows are separated by newlines.

71) reverse pyramid star pattern

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

```
int main() {
```

```
    int rows, i, j, space;
```

```
    printf("Enter the number of rows: ");
```

```
    scanf("%d", &rows);
```

```
    for (i = rows; i >= 1; i--) {
```

```
        for (space = 1; space <= rows - i; space++) {
```

```
            printf(" ");
```

```
        }
```

```
        for (j = 1; j <= 2 * i - 1; j++) {
```

```
            printf("*");
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```



```
input
Enter the number of rows: 6
*****
*****
*****
***
**
*
```

72) inverted hollow pyramid star pattern

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

```
int main() {
```

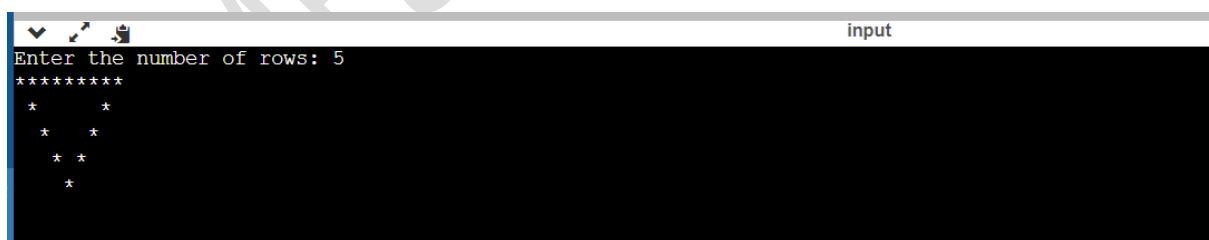
```
    int rows, i, j, space;
```

```
    printf("Enter the number of rows: ");
```

```

scanf("%d", &rows);
for (i = rows; i >= 1; i--) {
    for (space = 1; space <= rows - i; space++) {
        printf(" ");
    }
    if (i == 1 || i == rows) {
        for (j = 1; j <= 2 * i - 1; j++) {
            printf("*");
        }
    } else {
        printf("*");
        for (j = 1; j <= 2 * i - 3; j++) {
            printf(" ");
        }
        printf("*");
    }
    printf("\n");
}
return 0;
}

```



```

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

```

73) half diamond star pattern

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

```

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

```

```

printf("Enter the number of rows: ");
scanf("%d", &n);
for (i = 1; i <= n; i++) {
    for (j = 1; j <= i; j++) {
        printf("*");
    }
    printf("\n");
}
for (i = n - 1; i >= 1; i--) {
    for (j = 1; j <= i; j++) {
        printf("*");
    }
    printf("\n");
}

return 0;
}

```

```

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

```

74) mirrored diamond star pattern

```

#include <stdio.h>

int main() {
    int n, i, j, k;

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

    for (i = 1; i <= n; i++) {
        for (j = 1; j <= n - i; j++) {

```



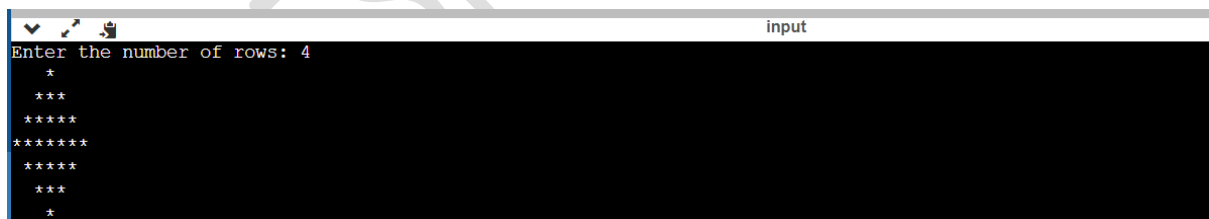
```

        printf(" ");
    }
    for (k = 1; k <= 2 * i - 1; k++) {
        printf("*");
    }
    printf("\n");
}

for (i = n - 1; i >= 1; i--) {
    for (j = 1; j <= n - i; j++) {
        printf(" ");
    }
    for (k = 1; k <= 2 * i - 1; k++) {
        printf("*");
    }
    printf("\n");
}

return 0;
}

```



The screenshot shows a terminal window with the title 'input'. The prompt 'Enter the number of rows: 4' is followed by the output of a C program. The output is a star pattern consisting of 4 rows: the first row has 1 star, the second has 3 stars, the third has 5 stars, and the fourth has 7 stars. The stars are centered, creating a diamond-like shape.

```

Enter the number of rows: 4
  *
 ***
*****
*****
  *

```

75) 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 n,i,j;
```

```
    printf ("enter the size of square: ");
```

```
    scanf ("%d",&n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        for (int j = 0; j < n; j++) {
```

```
            printf("1");
```

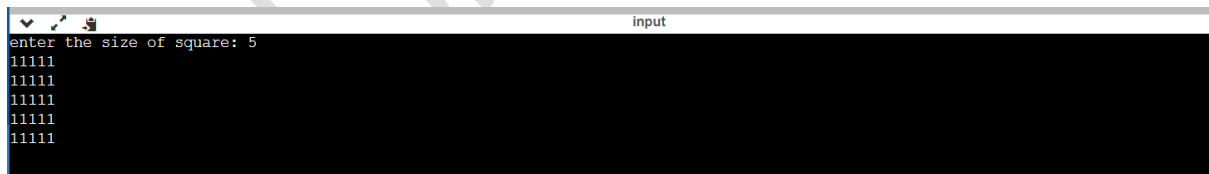
```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```



```
enter the size of square: 5
11111
11111
11111
11111
11111
11111
```

76) **11111**

00000

11111

00000

11111

```

#include <stdio.h>

int main() {
    int n;

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

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

    return 0;
}

```

```

input
01010
10101
01010
10101
...Program finished with exit code 0
Press ENTER to exit console.

```

77) 01010

01010

01010

01010

01010

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of rows: ");
```

```
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i++) {
```

```
        for (int j = 1; j <= n; j++) {
```

```
            if (j % 2 == 0) {
```

```
                printf("1");
```

```
            } else {
```

```
                printf("0");
```

```
            }
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```



```
input
Enter the number of rows: 5
01010
01010
01010
01010
01010
```

78) 11111

10001

10001

10001

11111

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of rows: ");
```

```
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i++) {
```

```
        for (int j = 1; j <= n; j++) {
```

```
            if (j == 1 || j == n || i == 1 || i == n) {
```

```
                printf("1");
```

```
            } else {
```

```
                printf("0");
```

```
            }
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```

```
input
Enter the number of rows: 5
11111
10001
10001
10001
11111
```

79) 10101

01010

10101

01010

10101

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of rows: ");
```

```
    scanf("%d", &n);
```

```
    for (int i = 1; i <= n; i++) {
```

```
        if (i % 2 == 1) {
```

```
            for (int j = 1; j <= n; j++) {
```

```
                if (j % 2 == 1) {
```

```
                    printf("1");
```

```
                } else {
```

```
                    printf("0");
```

```
                }
```

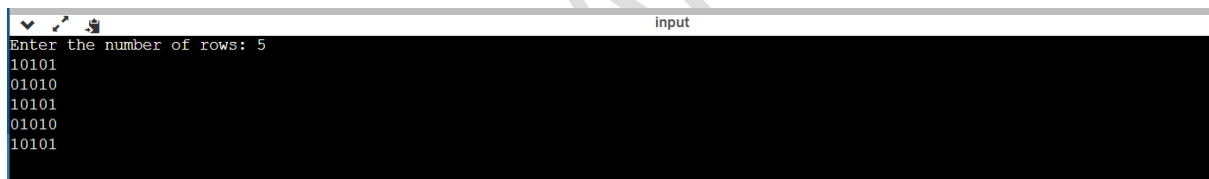
```
            }
```

```

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

    return 0;
}

```



```

input
Enter the number of rows: 5
10101
01010
10101
01010
10101

```

If...Else

80) Write a C program to find maximum between two numbers.

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

```

int main() {
    int num1, num2;

```

```

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

printf("Enter the second number: ");
scanf("%d", &num2);
if (num1 > num2) {
    printf("Maximum: %d\n", num1);
} else if (num2 > num1) {
    printf("Maximum: %d\n", num2);
} else {
    printf("Both numbers are equal.\n");
}

return 0;
}

```



```

input
Enter the first number: 3
Enter the second number: 4
Maximum: 4

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

```

81) Write a C program to find maximum between three numbers.

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

```

int main() {
    int num1, num2, num3;

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

    printf("Enter the second number: ");

```



```

scanf("%d", &num2);

printf("Enter the third number: ");
scanf("%d", &num3);
if (num1 >= num2) {
    if (num1 >= num3) {
        printf("Maximum: %d\n", num1);
    } else {
        printf("Maximum: %d\n", num3);
    }
} else {
    if (num2 >= num3) {
        printf("Maximum: %d\n", num2);
    } else {
        printf("Maximum: %d\n", num3);
    }
}

return 0;
}

```



```

input
Enter the first number: 4
Enter the second number: 5
Enter the third number: 2
Maximum: 5
...Program finished with exit code 0
Press ENTER to exit console

```

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

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

```
int main() {
```

```

int num;

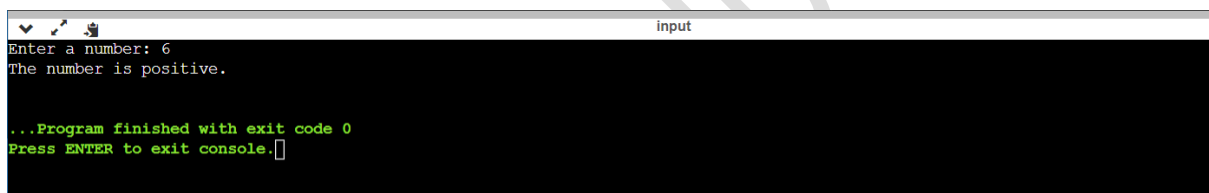
printf("Enter a number: ");

scanf("%d", &num);

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

return 0;
}

```



```

input
Enter a number: 6
The number is positive.

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

```

83) Write a C program to check whether a number is divisible by 5 and 11 or not

```

#include <stdio.h>

int main() {
    int num;

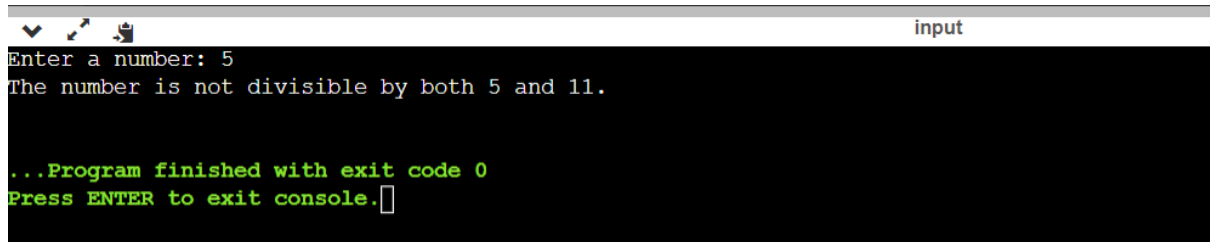
    printf("Enter a number: ");

    scanf("%d", &num);

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

```

```
}  
  
return 0;  
  
}
```

A screenshot of a terminal window with a dark background. The title bar at the top is light gray and contains the word "input" on the right. The terminal shows the following text: "Enter a number: 5" on the first line, "The number is not divisible by both 5 and 11." on the second line, and on the third line, "...Program finished with exit code 0" and "Press ENTER to exit console." followed by a cursor. There are some small icons in the top left of the terminal window.

```
input  
Enter a number: 5  
The number is not divisible by both 5 and 11.  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

84) Write a C program to check whether a number is even or odd

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

```
int main() {  
    int num;  
    printf("Enter a number: ");  
    scanf("%d", &num);  
    if (num % 2 == 0) {  
        printf("The number is even.\n");  
    } else {  
        printf("The number is odd.\n");  
    }  
  
    return 0;  
}
```

```
input
Enter a number: 8
The number is even.

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

85) write a C program to check whether a year is leap year or not.

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

```
int main() {
    int year;
    printf("Enter a year: ");
    scanf("%d", &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;
}
```

```
input
Enter a year: 2002
2002 is not a leap year.

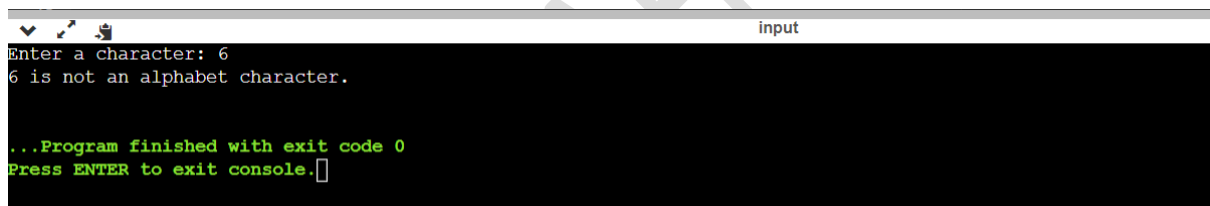
...Program finished with exit code 0
Press ENTER to exit console.
```

86) Write a C program to check whether a character is alphabet or not

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

```
#include <ctype.h>
```

```
int main() {  
    char character;  
    printf("Enter a character: ");  
    scanf(" %c", &character);  
    if (isalpha(character)) {  
        printf("%c is an alphabet character.\n", character);  
    } else {  
        printf("%c is not an alphabet character.\n", character);  
    }  
  
    return 0;  
}
```



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

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

```
#include <ctype.h>
```

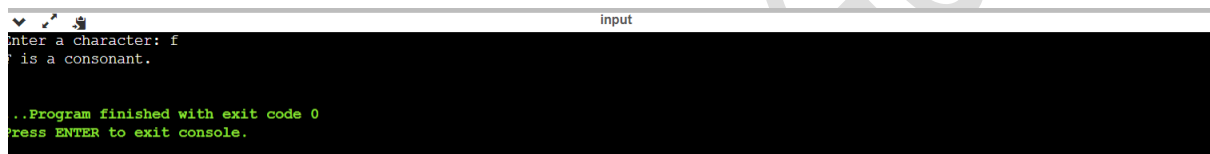
```
int main() {  
    char character;  
    printf("Enter a character: ");  
    scanf(" %c", &character);  
    character = toupper(character);  
    if (isalpha(character)) {
```

```

    if (character == 'A' || character == 'E' || character == 'I' || character == 'O' || character == 'U') {
        printf("%c is a vowel.\n", character);
    } else {
        printf("%c is a consonant.\n", character);
    }
} else {
    printf("Invalid input. Please enter an alphabet character.\n");
}

return 0;
}

```



```

input
Enter a character: f
f is a consonant.

..Program finished with exit code 0
press ENTER to exit console.

```

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

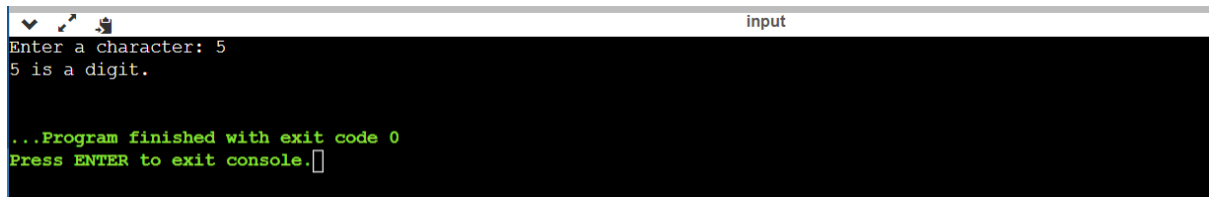
```

#include <stdio.h>
#include <ctype.h>

int main() {
    char character;
    printf("Enter a character: ");
    scanf("%c", &character);
    if (isalpha(character)) {
        printf("%c is an alphabet character.\n", character);
    } else if (isdigit(character)) {
        printf("%c is a digit.\n", character);
    } else {
        printf("%c is a special character.\n", character);
    }
}

```

```
}  
  
return 0;  
  
}
```

A screenshot of a terminal window with a black background and white text. The window title is 'input'. The text inside shows the program's execution: 'Enter a character: 5', '5 is a digit.', and a green message '...Program finished with exit code 0' followed by 'Press ENTER to exit console.'.

```
input  
Enter a character: 5  
5 is a digit.  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

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

```
#include <stdio.h>  
  
#include <ctype.h>  
  
int main() {  
    char character;  
    printf("Enter a character: ");  
    scanf(" %c", &character);  
    if (isupper(character)) {  
        printf("%c is an uppercase alphabet.\n", character);  
    } else if (islower(character)) {  
        printf("%c is a lowercase alphabet.\n", character);  
    } else {  
        printf("%c is not an alphabet character.\n", character);  
    }  
  
    return 0;  
}
```

```
input
Enter a character: k
k is a lowercase alphabet.

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

90) 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("Monday\n");
            break;
        case 2:
            printf("Tuesday\n");
            break;
        case 3:
            printf("Wednesday\n");
            break;
        case 4:
            printf("Thursday\n");
            break;
        case 5:
            printf("Friday\n");
            break;
        case 6:
```



```

        printf("Saturday\n");

        break;

    case 7:

        printf("Sunday\n");

        break;

    default:

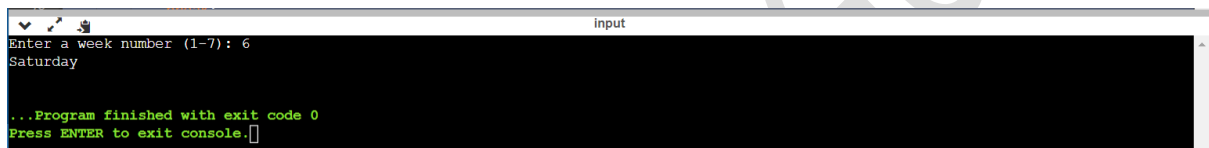
        printf("Invalid week number. Please enter a number between 1 and 7.\n");

    }

    return 0;

}

```



```

input
Enter a week number (1-7): 6
Saturday

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

```

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

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

```

int main() {

    int monthNumber;

    printf("Enter a month number (1-12): ");

    scanf("%d", &monthNumber);

    switch (monthNumber) {

        case 1: case 3: case 5: case 7: case 8: case 10: case 12:

            printf("Number of days in this month: 31\n");

            break;

        case 4: case 6: case 9: case 11:

            printf("Number of days in this month: 30\n");

            break;

```

```

case 2:

    printf("Number of days in this month: 28 or 29 (leap year)\n");

    break;

default:

    printf("Invalid month number. Please enter a number between 1 and 12.\n");

}

return 0;

}

```

```

input
Enter a month number (1-12): 8
Number of days in this month: 31

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

```

92) Write a C program to count total number of notes in given amount

```

#include <stdio.h>

int main() {

    int amount;

    int notes[] = {2000, 500, 200, 100, 50, 20, 10, 5, 1};

    int count[9] = {0};

    printf("Enter the amount: ");

    scanf("%d", &amount);

    if (amount <= 0) {

        printf("Invalid amount. Please enter a positive amount.\n");

    } else {

        printf("Number of notes for the given amount:\n");
    }
}

```

```

for (int i = 0; i < 9; i++) {
    if (amount >= notes[i]) {
        count[i] = amount / notes[i];
        amount = amount % notes[i];
        printf("%d notes of %d\n", count[i], notes[i]);
    }
}

return 0;
}

```

```

input
Enter the amount: 700
Number of notes for the given amount:
1 notes of 500
1 notes of 200

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

```

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

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

```

int main() {
    float angle1, angle2, angle3;
    printf("Enter the first angle: ");
    scanf("%f", &angle1);

    printf("Enter the second angle: ");
    scanf("%f", &angle2);

    printf("Enter the third angle: ");

```

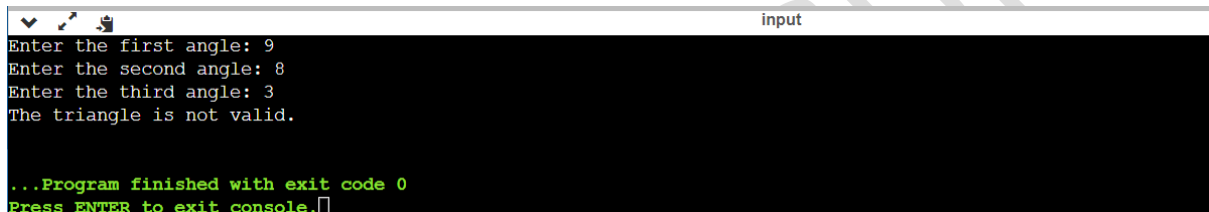
```

scanf("%f", &angle3);

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

return 0;
}

```



```

input
Enter the first angle: 9
Enter the second angle: 8
Enter the third angle: 3
The triangle is not valid.

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

```

94) 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 length of the first side: ");
    scanf("%f", &side1);

    printf("Enter the length of the second side: ");
    scanf("%f", &side2);

    printf("Enter the length of the third side: ");
    scanf("%f", &side3);

    if ((side1 + side2 > side3) && (side1 + side3 > side2) && (side2 + side3 > side1)) {


```

```

        printf("The triangle is valid.\n");
    } else {
        printf("The triangle is not valid.\n");
    }

    return 0;
}

```



```

input
Enter the length of the first side: 5
Enter the length of the second side: 3
Enter the length of the third side: 4
The triangle is valid.
...Program finished with exit code 0
Press ENTER to exit console.

```

95) 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 length of the first side: ");
    scanf("%f", &side1);

    printf("Enter the length of the second side: ");
    scanf("%f", &side2);

    printf("Enter the length of the third side: ");
    scanf("%f", &side3);

    if ((side1 + side2 > side3) && (side1 + side3 > side2) && (side2 + side3 > side1)) {
        if (side1 == side2 && side2 == side3) {
            printf("It's an equilateral triangle.\n");
        } else if (side1 == side2 || side1 == side3 || side2 == side3) {
            printf("It's an isosceles triangle.\n");

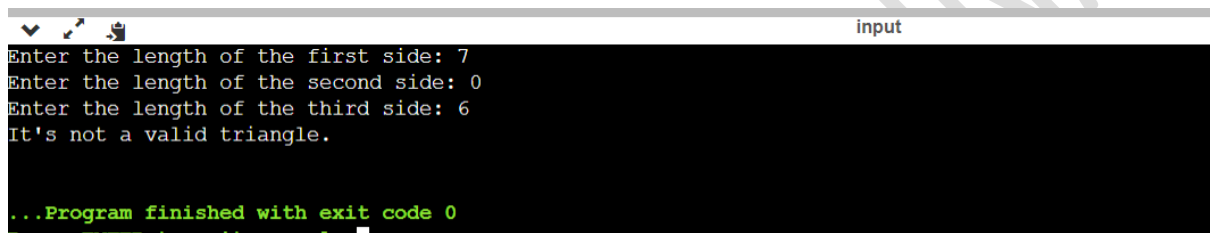
```

```

    } else {
        printf("It's a scalene triangle.\n");
    }
} else {
    printf("It's not a valid triangle.\n");
}

return 0;
}

```



```

input
Enter the length of the first side: 7
Enter the length of the second side: 0
Enter the length of the third side: 6
It's not a valid triangle.

...Program finished with exit code 0

```

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

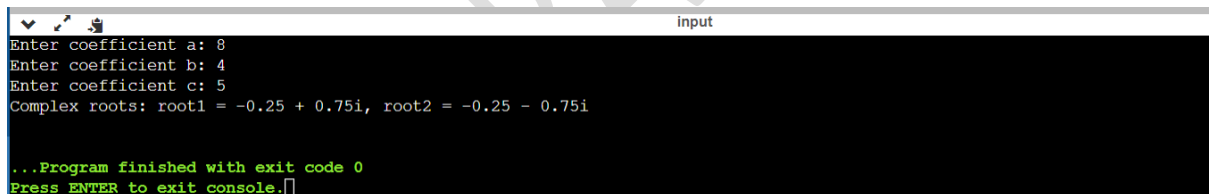
```

```

if (discriminant > 0) {
    root1 = (-b + sqrt(discriminant)) / (2 * a);
    root2 = (-b - sqrt(discriminant)) / (2 * a);
    printf("Two distinct real roots: root1 = %.2lf, root2 = %.2lf\n", root1, root2);
} else if (discriminant == 0) {
    root1 = root2 = -b / (2 * a);
    printf("Two equal real roots: root1 = root2 = %.2lf\n", root1);
} else {
    double realPart = -b / (2 * a);
    double imaginaryPart = sqrt(-discriminant) / (2 * a);
    printf("Complex roots: root1 = %.2lf + %.2lfi, root2 = %.2lf - %.2lfi\n", realPart, imaginaryPart,
    realPart, imaginaryPart);
}

return 0;
}

```



```

input
Enter coefficient a: 8
Enter coefficient b: 4
Enter coefficient c: 5
Complex roots: root1 = -0.25 + 0.75i, root2 = -0.25 - 0.75i
...Program finished with exit code 0
Press ENTER to exit console.

```

97) Write a C program to calculate profit or loss

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

```

int main() {
    float costPrice, sellingPrice, profitOrLoss;

    printf("Enter the cost price: ");

    scanf("%f", &costPrice);

    printf("Enter the selling price: ");

```

```

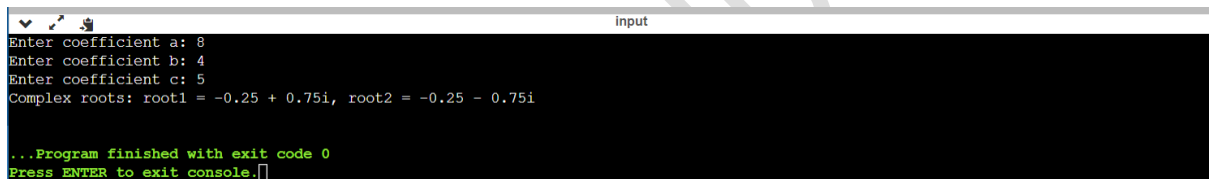
scanf("%f", &sellingPrice);

profitOrLoss = sellingPrice - costPrice;

if (profitOrLoss > 0) {
    printf("You made a profit of $%.2f\n", profitOrLoss);
} else if (profitOrLoss < 0) {
    printf("You incurred a loss of $%.2f\n", -profitOrLoss);
} else {
    printf("You broke even; no profit or loss.\n");
}

return 0;
}

```



```

input
Enter coefficient a: 8
Enter coefficient b: 4
Enter coefficient c: 5
Complex roots: root1 = -0.25 + 0.75i, root2 = -0.25 - 0.75i

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

```

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

Percentage $\geq 90\%$: Grade A

Percentage $\geq 80\%$: Grade B

Percentage $\geq 70\%$: Grade C

Percentage $\geq 60\%$: Grade D

Percentage $\geq 40\%$: Grade E

Percentage < 40% : Grade F

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

```
int main() {  
    float physics, chemistry, biology, mathematics, computer;  
    float totalMarks, percentage;  
    char grade;  
    printf("Enter marks for Physics: ");  
    scanf("%f", &physics);  
    printf("Enter marks for Chemistry: ");  
    scanf("%f", &chemistry);  
    printf("Enter marks for Biology: ");  
    scanf("%f", &biology);  
    printf("Enter marks for Mathematics: ");  
    scanf("%f", &mathematics);  
    printf("Enter marks for Computer: ");  
    scanf("%f", &computer);  
    totalMarks = physics + chemistry + biology + mathematics + computer;  
    percentage = (totalMarks / 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';
    }

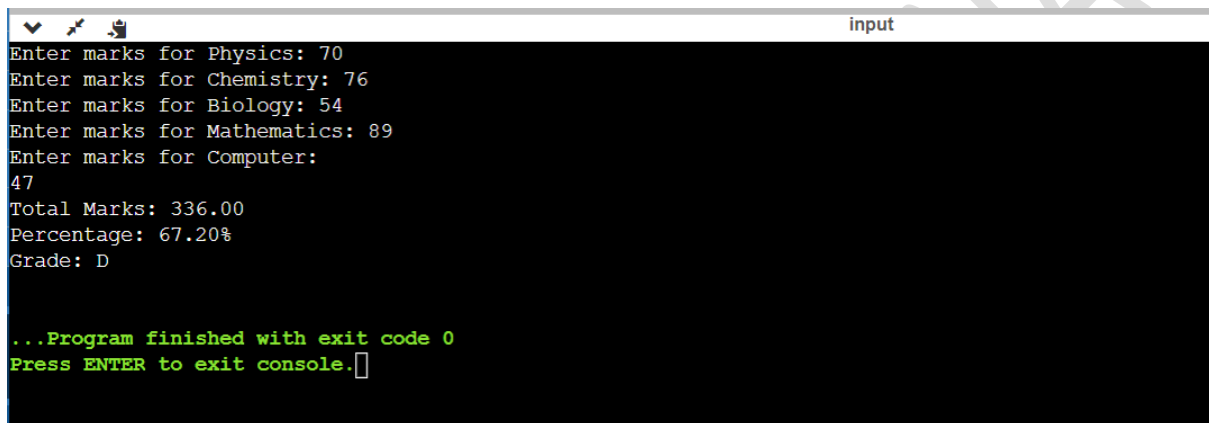
    printf("Total Marks: %.2f\n", totalMarks);

    printf("Percentage: %.2f%%\n", percentage);

    printf("Grade: %c\n", grade);

    return 0;
}

```



```

input
Enter marks for Physics: 70
Enter marks for Chemistry: 76
Enter marks for Biology: 54
Enter marks for Mathematics: 89
Enter marks for Computer:
47
Total Marks: 336.00
Percentage: 67.20%
Grade: D

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

```

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

Basic Salary \leq 10000 : HRA = 20%, DA = 80%

Basic Salary \leq 20000 : HRA = 25%, DA = 90%

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

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

```

int main() {

    float basicSalary, grossSalary;

    float hra, da;

```

```

printf("Enter the basic salary: ");
scanf("%f", &basicSalary);
if (basicSalary <= 10000) {
    hra = 0.2 * basicSalary;
    da = 0.8 * basicSalary;
} else if (basicSalary <= 20000) {
    hra = 0.25 * basicSalary;
    da = 0.9 * basicSalary;
} else {
    hra = 0.3 * basicSalary;
    da = 0.95 * basicSalary;
}
grossSalary = basicSalary + hra + da;
printf("Gross Salary: %.2f\n", grossSalary);

return 0;
}

```



```

input
Enter the basic salary: 70000
Gross Salary: 157500.00
...Program finished with exit code 0
Press ENTER to exit console.

```

100) Write a C program to input electricity unit charges and calculate total electricity bill according to the given condition:

For first 50 units Rs. 0.50/unit

For next 100 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 unitCharges, totalBill;
```

```
    printf("Enter the electricity unit charges: ");
```

```
    scanf("%f", &unitCharges);
```

```
    if (unitCharges <= 50) {
```

```
        totalBill = unitCharges * 0.50;
```

```
    } else if (unitCharges <= 150) {
```

```
        totalBill = (50 * 0.50) + ((unitCharges - 50) * 0.75);
```

```
    } else if (unitCharges <= 250) {
```

```
        totalBill = (50 * 0.50) + (100 * 0.75) + ((unitCharges - 150) * 1.20);
```

```
    } else {
```

```
        totalBill = (50 * 0.50) + (100 * 0.75) + (100 * 1.20) + ((unitCharges - 250) * 1.50);
```


```
    }
```

```
    totalBill += 0.20 * totalBill;
```

```
    printf("Total Electricity Bill: Rs. %.2f\n", totalBill);
```

```
    return 0;
```

```
}
```



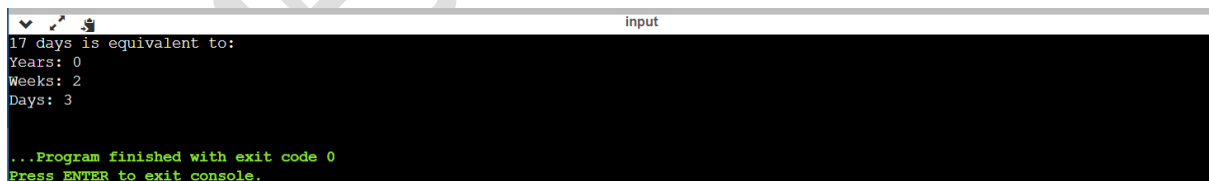
```
input
Enter the electricity unit charges: 50
Total Electricity Bill: Rs. 30.00

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

101) Write a C program to convert specified days into years, weeks and days

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

```
int main() {  
    int days, years, weeks, remainingDays;  
    printf("Enter the number of days: ");  
    scanf("%d", &days);  
    years = days / 365;  
    weeks = (days % 365) / 7;  
    remainingDays = days - (years * 365) - (weeks * 7);  
    printf("%d days is equivalent to:\n", days);  
    printf("Years: %d\n", years);  
    printf("Weeks: %d\n", weeks);  
    printf("Days: %d\n", remainingDays);  
  
    return 0;  
}
```



```
input  
17 days is equivalent to:  
Years: 0  
Weeks: 2  
Days: 3  
...Program finished with exit code 0  
Press ENTER to exit console.
```

ARRAY IN C

102) 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 values: ");
```

```
    scanf("%d", &n);
```

```
    int arr[n];
```

```
    printf("Enter %d values:\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
    printf("Values in reverse order:\n");
```

```
    for (int i = n - 1; i >= 0; i--) {
```

```
        printf("%d\n", arr[i]);
```

```
    }
```

```
    return 0;
```

```
}
```

```
input
Enter the number of values: 2
Enter 2 values:
6
7
Values in reverse order:
7
6
```

103) Write a program in C to find the sum of all elements of the array

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

```
int main() {
    int n;

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

    int arr[n];

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

    int sum = 0;
    for (int i = 0; i < n; i++) {
        sum += arr[i];
    }

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

    return 0;
}
```

```
input
Enter the number of elements in the array: 3
Enter 3 elements:
4
5
8
Sum of all elements: 17
```

104) Write a program in C to copy the elements of one array into another array

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in the array: ");
```

```
    scanf("%d", &n);
```

```
    int sourceArray[n];
```

```
    int destinationArray[n];
```

```
    printf("Enter %d elements for the source array:\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &sourceArray[i]);
```

```
    }
```

```
    for (int i = 0; i < n; i++) {
```

```
        destinationArray[i] = sourceArray[i];
```

```
    }
```

```
    printf("Elements in the destination array after copying:\n");
```

```
    for (int i = 0; i < n; i++) {
```

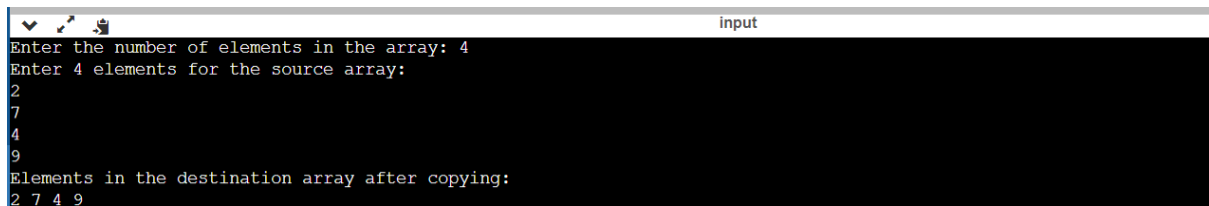


```

        printf("%d ", destinationArray[i]);
    }
    printf("\n");

    return 0;
}

```



```

input
Enter the number of elements in the array: 4
Enter 4 elements for the source array:
2
7
4
9
Elements in the destination array after copying:
2 7 4 9

```

105) Write a program in C to count the total number of duplicate elements in an array

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

```

int main() {
    int n;

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

    int arr[n];

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

```

```

int duplicateCount = 0;
for (int i = 0; i < n; i++) {
    for (int j = i + 1; j < n; j++) {
        if (arr[i] == arr[j]) {
            duplicateCount++;
            break;
        }
    }
}

printf("Total number of duplicate elements: %d\n", duplicateCount);

return 0;
}

```



```

input
Enter the number of elements in the array: 4
Enter 4 elements:
22
7
6
7
Total number of duplicate elements: 1

```

106) Write a program in C to find the maximum and minimum elements in an array

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in the array: ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 0) {
```

```
    printf("Invalid array size. Please enter a positive integer.\n");
    return 1;
}

int arr[n];

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

int max = arr[0];
int min = arr[0];

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

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

return 0;
}
```

```
input
Enter the number of elements in the array: 3
Enter 3 elements:
5
8
3
Maximum element: 8
Minimum element: 3
```

107) Write a C program to sort the elements of an array in descending order

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

```
void selectionSort(int arr[], int n) {
```

```
    for (int i = 0; i < n - 1; i++) {
```

```
        int max_index = i;
```

```
        for (int j = i + 1; j < n; j++) {
```

```
            if (arr[j] > arr[max_index]) {
```

```
                max_index = j;
```

```
            }
```

```
        }
```

```
        if (max_index != i) {
```

```
            int temp = arr[i];
```

```
            arr[i] = arr[max_index];
```

```
            arr[max_index] = temp;
```

```
        }
```

```
    }
```

```
}
```

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in the array: ");
```

```
scanf("%d", &n);

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


int arr[n];

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

selectionSort(arr, n);

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

return 0;
}
```



```
Enter the number of elements in the array: 3
Enter 3 elements:
4
9
5
Sorted array in descending order:
9 5 4
```

108) Write a program in C to separate odd and even integers into separate arrays

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in the array: ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 0) {
```

```
        printf("Invalid array size. Please enter a positive integer.\n");
```

```
        return 1;
```

```
    }
```

```
    int arr[n];
```

```
    printf("Enter %d elements:\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
    int oddArray[n];
```

```
    int evenArray[n];
```

```
    int oddCount = 0;
```

```
    int evenCount = 0;
```

```

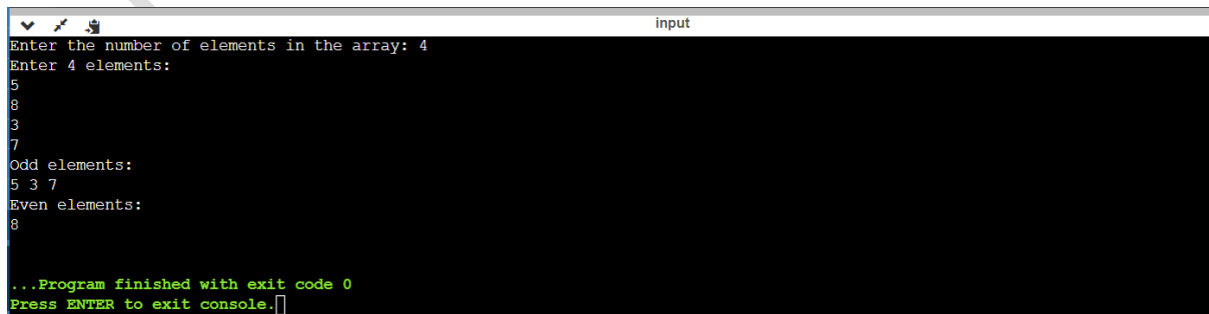
for (int i = 0; i < n; i++) {
    if (arr[i] % 2 == 0) {
        evenArray[evenCount] = arr[i];
        evenCount++;
    } else {
        oddArray[oddCount] = arr[i];
        oddCount++;
    }
}

printf("Odd elements:\n");
for (int i = 0; i < oddCount; i++) {
    printf("%d ", oddArray[i]);
}
printf("\n");

printf("Even elements:\n");
for (int i = 0; i < evenCount; i++) {
    printf("%d ", evenArray[i]);
}
printf("\n");

return 0;
}

```



```

input
Enter the number of elements in the array: 4
Enter 4 elements:
5
8
3
7
Odd elements:
5 3 7
Even elements:
8
...Program finished with exit code 0
Press ENTER to exit console.

```

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

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in each sorted array: ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 0) {
```

```
        printf("Invalid array size. Please enter a positive integer.\n");
```

```
        return 1;
```

```
    }
```

```
    // Declare two arrays to store the sorted elements
```

```
    int arr1[n];
```

```
    int arr2[n];
```

```
    int mergedArray[2 * n];
```

```
    printf("Enter %d elements for the first sorted array (ascending order):\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr1[i]);
```

```
    }
```

```
    printf("Enter %d elements for the second sorted array (ascending order):\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr2[i]);
```

```
    }
```



```

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

while (i < n) {
    mergedArray[k] = arr1[i];
    i++;
    k++;
}

while (j < n) {
    mergedArray[k] = arr2[j];
    j++;
    k++;
}

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

printf("\n");

return 0;
}

```

```
input
Enter the number of elements in each sorted array: 3
Enter 3 elements for the first sorted array (ascending order):
9
5
2
Enter 3 elements for the second sorted array (ascending order):
8
4
1
Merged array in ascending order:
8 4 1 9 5 2
```

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

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

```
int main() {
```

```
    int n;
```

```
    printf("Enter the number of elements in each sorted array: ");
```

```
    scanf("%d", &n);
```

```
    if (n <= 0) {
```

```
        printf("Invalid array size. Please enter a positive integer.\n");
```

```
        return 1;
```

```
    }
```

```
    int arr1[n];
```

```
    int arr2[n];
```

```
    int mergedArray[2 * n];
```

```
    printf("Enter %d elements for the first sorted array (descending order):\n", n);
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr1[i]);
```

```
    }
```

```
    printf("Enter %d elements for the second sorted array (descending order):\n", n);
```

```
for (int i = 0; i < n; i++) {  
    scanf("%d", &arr2[i]);  
}  
  
int i = 0, j = 0, k = 0;  
  
while (i < n && j < n) {  
    if (arr1[i] >= arr2[j]) {  
        mergedArray[k] = arr1[i];  
        i++;  
    } else {  
        mergedArray[k] = arr2[j];  
        j++;  
    }  
    k++;  
}
```

```
while (i < n) {  
    mergedArray[k] = arr1[i];  
    i++;  
    k++;  
}
```

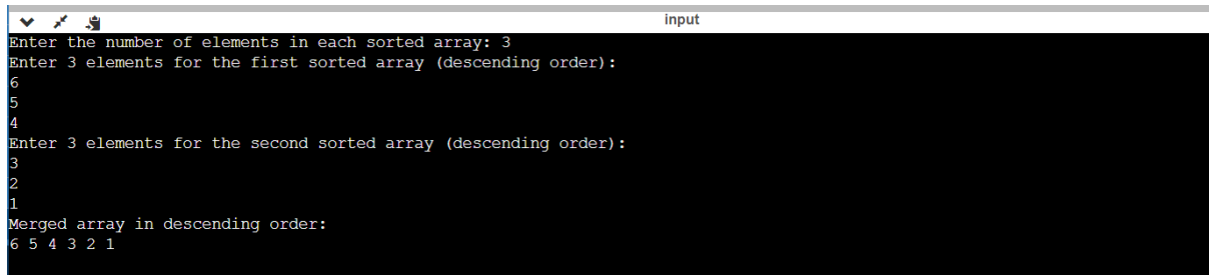
```
while (j < n) {  
    mergedArray[k] = arr2[j];  
    j++;  
    k++;  
}
```

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

```
printf("\n");
```

```
return 0;
```

```
}
```



The screenshot shows a terminal window with a title bar containing a heart icon, a magnifying glass icon, and a terminal icon, followed by the word "input". The terminal output is as follows:

```
Enter the number of elements in each sorted array: 3
Enter 3 elements for the first sorted array (descending order):
6
5
4
Enter 3 elements for the second sorted array (descending order):
3
2
1
Merged array in descending order:
6 5 4 3 2 1
```