**LAB FILE**

**INTRODUCTION TO C**

**S**

***BATCH: 2023-2027***

***COURSE: BCA(HONS) AI &* DS**

SUBMITTED BY: SURBHI

**STUDENT NAME: SURBHI** SUBMITTED TO:

**STUDENT ID:** 23152395 **MR. RISHI**

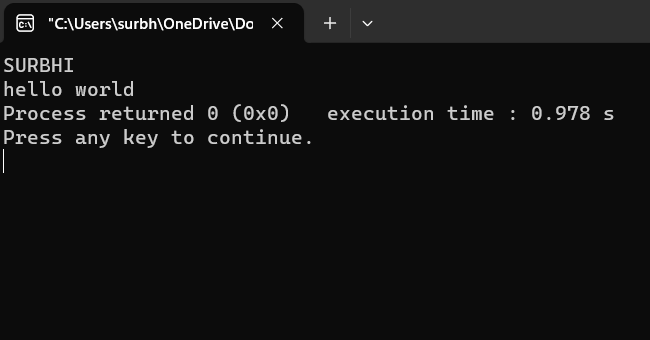
**KUMAR SIR**

**&Pooja Mam..**

INDEX

|  |  |  |  |
| --- | --- | --- | --- |
| ***Program Name*** | ***Page NO.*** | ***F. Sign*** | ***Remark*** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. WAP for hello world or this is my first C Program :

 #include<stdio.h>

int main()

{

printf("SURBHI\n");

printf("hello world");

return 0;

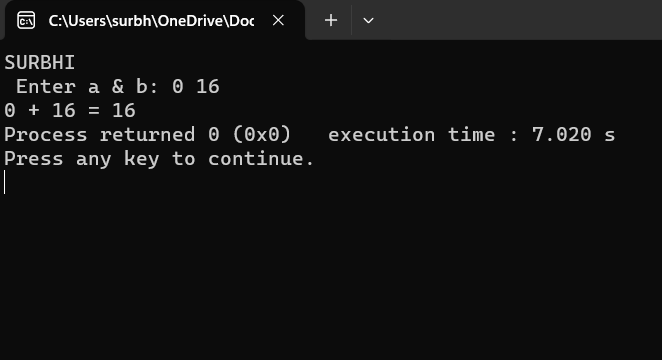
## }

**2.WAP to add two numbers:**

#include<stdio.h>

int main()

{

int a, b, sum;

printf("SURBHI\n");

printf(" Enter a & b:");

scanf("%d, &a, &b");

// calculate the sum

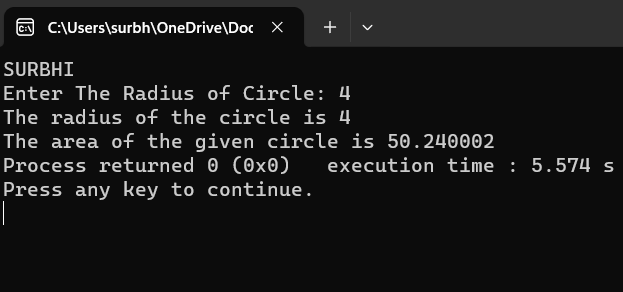
sum = a + b;

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

return 0;

}

1. **WAP to find area of circle:**

#include <stdio.h>

int main() {

float pie = 3.14;

int radius;

printf("SURBHI\n");

printf("Enter The Radius of Circle:");

scanf("%d",&radius);

printf("The radius of the circle is %d\n" , radius);

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

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

return 0;

}

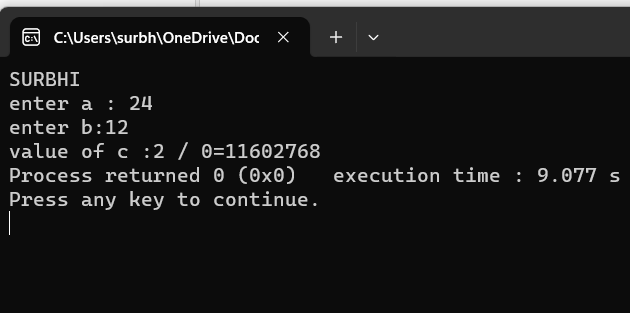
1. **WAP to divide two numbers:**

#include<stdio.h>

int main()

{

int a, b, c;

 printf("SURBHI\n");

printf("enter a :");

scanf("%d",&a);

printf("enter b:");

scanf("%d",&b);

//divide two numbers

c= a/b;

printf("value of c :%d / %d=%d ",c);

return 0;

}

1. **WAP to print ASCII value:**

#include<stdio.h>

int main()

{

char a;

printf("SURBHI\n");

printf("enter a character:");

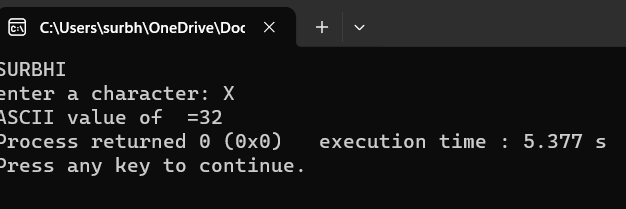
scanf("%c",&a);

//%d display the integer value of character

//%c display the actual character value

printf("ASCII value of %c=%d",a,a);

return 0;

}

1. **WAP to multiply floating point numbers:**

#include<stdio.h>

int main()

{

float a,b,c;

printf("SURBHI\n");

printf("enter a :");

scanf("%f",&a);

printf("enter b:");

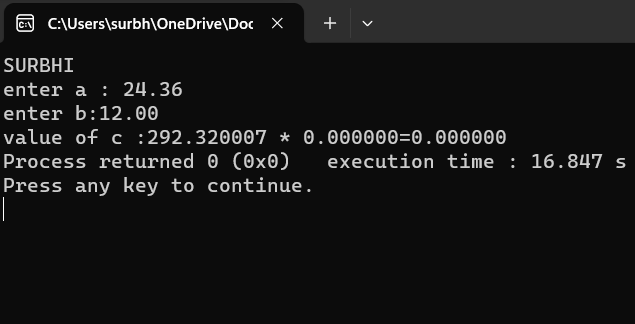
scanf("%f",&b);

//multiply two numbers

c= a\*b;

printf("value of c :%f \* %f=%f ",c);

return 0;

}

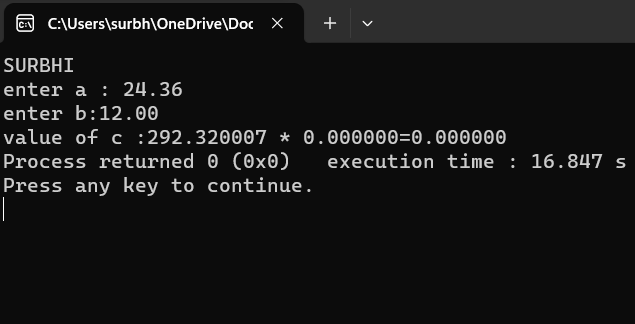
1. **WAP to SWAP two variables number by without using third variable:**

#include<stdio.h>

int main()

{

int a,b,c;

 printf("SURBHI\n");

printf("enter value of a");

scanf("%d",&a);

printf("\n enter value of b");

scanf("%d",&b);

printf("enter value of c:");

scanf("%d",&c);

// using a temporary variable

// store the value of a in temporary variable

int temp = a;

// assign the value of b to a

a = a+b;

b = a-b;

a=a-b;

printf("\n after swapping a = %d, b = %d,c = %d", a,b,c);

return 0 ;

}

1. **WAP to SWAP two variables number with using third variables:**

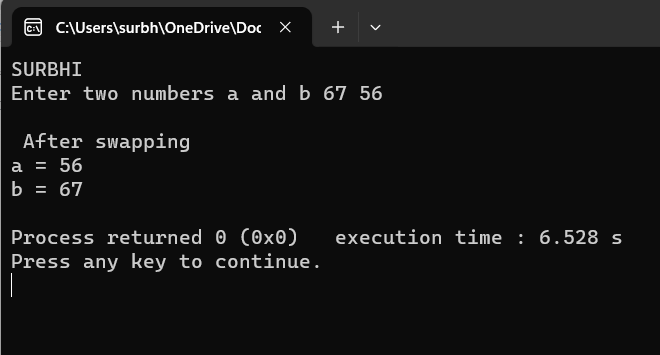
#include<stdio.h>

int main(void)

{

   int a, b, temp;

printf(“SURBHI\n”);



   printf("Enter two numbers a and b ");

   scanf("%d %d", &a, &b);

   // Swap logic

   temp = a;

   a = b;

   b = temp;

  printf("\n After swapping \na = %d\nb = %d\n", a, b);

  return 0;

}

1. **WAP to SWAP three variable numbers without using third variables:**

#include<stdio.h>

int main()

{

int a,b,c;

printf("SURBHI\n");

printf("enter value of a");

scanf("%d",&a);

printf("\n enter value of b");

scanf("%d",&b);

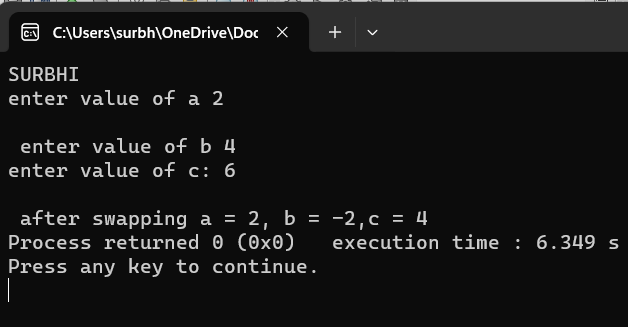
printf("enter value of c:");

scanf("%d",&c);

// using a temporary variable

// store the value of a in temporary variable

int temp = a;

 // assign the value of b to a

a = a +b+ c;

b = a-b-c;

c=a-b-c;

printf("\n after swapping a = %d, b = %d,c = %d", a,b,c);

return 0 ;

}

**10.WAP to find the area of rectangle:**

#include <stdio.h>

int main() {

float length, width, area;

printf("SURBHI\n");

printf("Enter the length of the rectangle: ");

scanf("%f", &length);

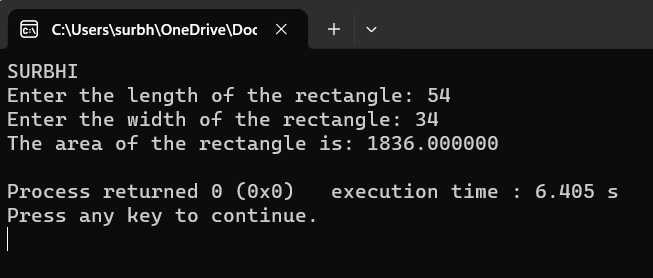
printf("Enter the width of the rectangle: ");

scanf("%f", &width);

area = length \* width;

printf("The area of the rectangle is: %f\n", area);

return 0;

}

**11. WAP to find area of square:**

#include<stdio.h>

int main()

{

printf("SURBHI\n");

int side,area;

printf("enter the side");

scanf("%d",&side);

//Area of square

area = side \* side;

printf("area is : %d",area);

return 0;

}

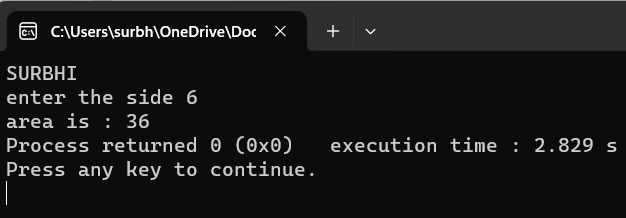
**12. WAP to find Area and Volume of Cube:**

#include<stdio.h>

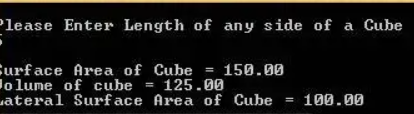
int main()

{

float l, SA ,Volume, LSA;

 printf("\n Please Enter Length of any side of a Cube \n");

scanf(" %f ", &l);

 SA = 6 \* (l \* l);

Volume = l \* l \* l;

LSA = 4 \* (l \* l);

printf("\n Surface Area of Cube = %.2f", SA);

printf("\n Volume of cube = %.2f", Volume);

printf("\n Lateral Surface Area of Cube = %.2f", LSA);

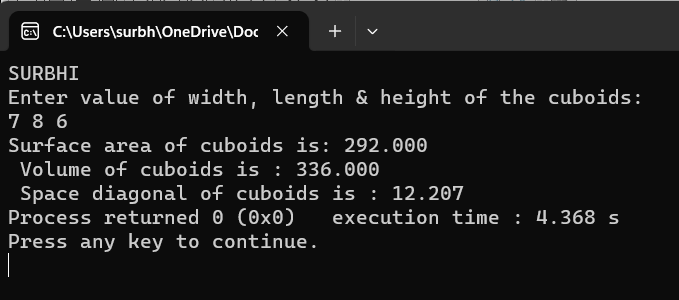
return 0;

}

13. WAP to find area and volume of cuboid:

#include <stdio.h>

#include <math.h>

int main()

{

float width, length, height;

float surfacearea, volume, space\_diagonal;

printf(“SURBHI\n”);

printf("Enter value of width, length & height of the cuboids:**\n**");

scanf("%f%f%f", &width, &length, &height);

surfacearea = 2 \*(width \* length + length \* height +

height \* width);

volume = width \* length \* height;

space\_diagonal = sqrt(width \* width + length \* length +

height \* height);

printf("Surface area of cuboids is: %.f", surfacearea);

printf("**\n** Volume of cuboids is : %.f", volume);

printf("**\n** Space diagonal of cuboids is : %.f", space\_diagonal);

return 0;

}

14. WAP to find the largest number using the Logical AND operator:

#include<stdio.h>

int main()

{

int n1,n2,n3;

printf("SURBHI\n");

printf("enter the n1:");

scanf("%f",&n1);

printf("enter the n2:");

scanf("%f",&n2);

printf("enter the n3:");

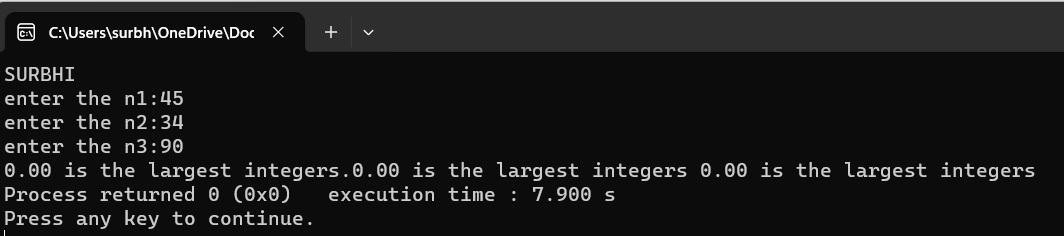
scanf("%f",&n3);

// if n1 is greater than both n2 & n3 , n1 is the largest //

if (n1>=n2 && n1>= n3);

printf("%.2f is the largest integers.", n1);

// if n2 is greater than both n1 & n2, n2 is the largest//

 if (n2>=n1 && n2>=n3);

printf("%.2f is the largest integers",n2);

// if n3 is greater than both n1 & n2 , n3 is the largest//

if (n3>=n1 && n3>=n2);

printf(" %.2f is the largest integers", n3);

return 0 ;

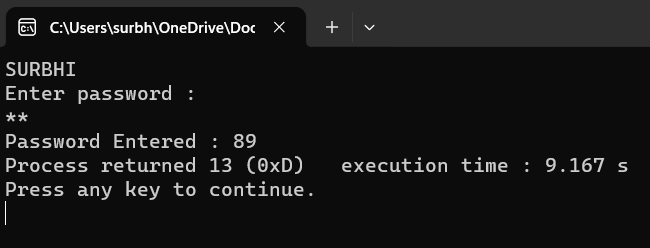
}

**15. WAP to validate the username and password entered by the user is correct or not using the predefined username and password:**

#include<stdio.h>

#include<conio.h>

void main() {

   char password[25], ch;

   int i;

printf(“SURBHI\n”);

  printf("Enter password : ");

   while (1) {

      if (i < 0) {

         i = 0;

      }

      ch = getch();

      if (ch == 13)

         break;

      if (ch == 8) /\*ASCII value of BACKSPACE\*/

      {

         putch('b');

         putch(NULL);

         putch('b');

         i--;

         continue;

      }

 password[i++] = ch;

      ch = '\*';

      putch(ch);

   }

password[i] = '\0';

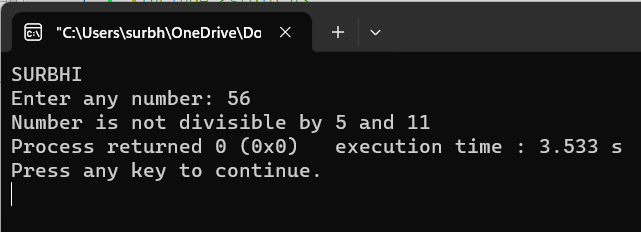
   printf("\nPassword Entered : %s", password);

getch();

}

**16. WAP for an integer number and to check whether it is divisible by 5 or 11 using OR logical operator:**

#include <stdio.h>

int main()

{

int num;

printf("SURBHI\n");

printf("Enter any number: ");

scanf("%d", &num);

if((num % 5 == 0) && (num % 11 == 0))

{

printf("Number is divisible by 5 and 11");

}

else

{

printf("Number is not divisible by 5 and 11");

}

return 0;

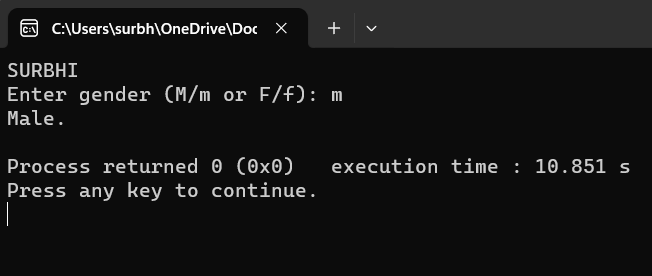
}

**17. 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("SURBHI\n");

printf("Enter gender (M/m or F/f): ");

scanf("%c",&gender);

switch(gender)

{

case 'M':

case 'm':

printf("Male.");

break;

case 'F':

case 'f':

printf("Female.");

break;

default:

printf("Unspecified Gender.");

}

printf("\n");

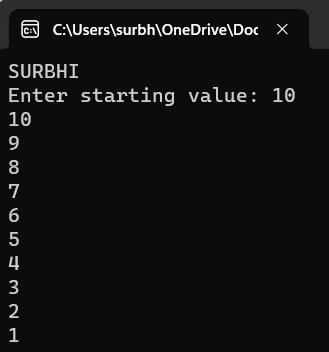
return 0;

}

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

#include <stdio.h>

int main()

{

int i, start;

printf("SURBHI\n");

printf("Enter starting value: ");

scanf("%d", &start);

for(i=start; i>=1; i--)

{

printf("%d\n", i);

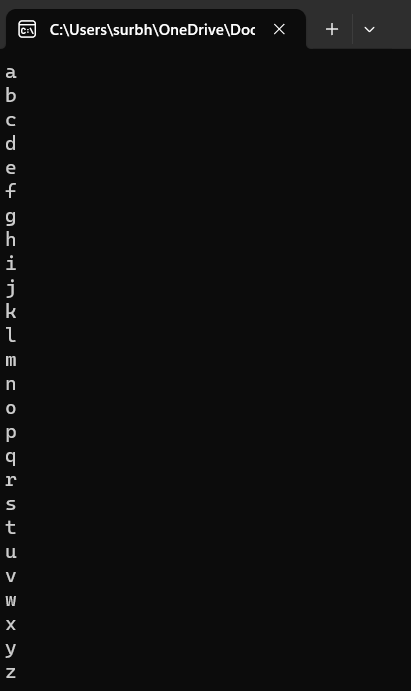
}

return 0;

}

**19. Write a C program to print all alphabets from a to z:**

#include <stdio.h>



int main()

{

char ch = 'a';

printf("SURBHI\n");

printf("Alphabets from a - z are: \n");

while(ch<='z')

{

printf("%c\n", ch);

ch++;

}

return 0;

}

**20.Write a C program to print all natural numbers from 1 to n:**

#**include** <stdio.h>

**int** main()

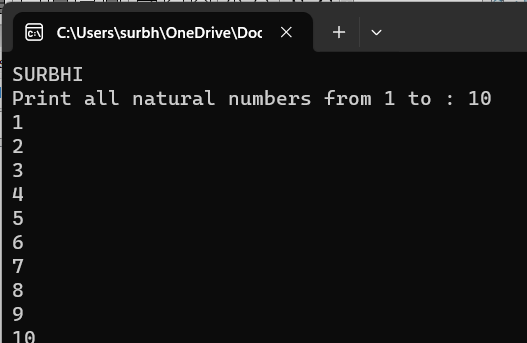
{

**int** i, end;

printf(“SURBHI\n);

printf("Print all natural numbers from 1 to : ");

scanf("%d", &end);



**while**(i<=end)

{

printf("%d\n", i);

i++;

}

**return** 0;

}

**21. program to print all even numbers between 1 to 100:**

#include<stdio.h>

int main(){

printf("SURBHI\n");

for (int i = 2; i <= 100; i++){

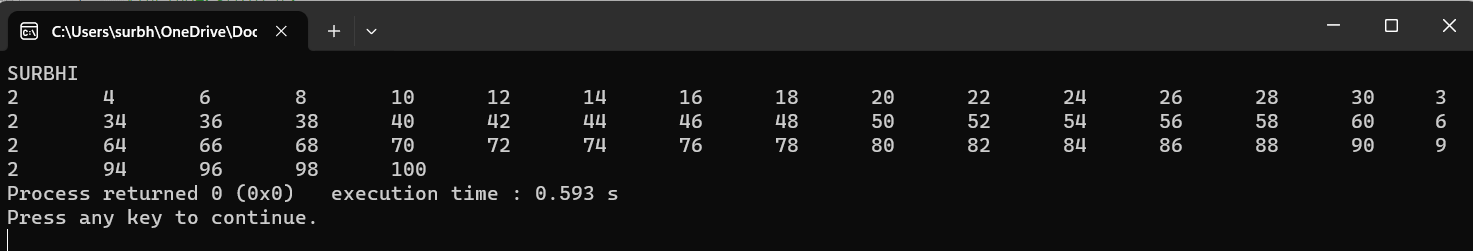
if (i % 2 == 0){

printf("%d\t", i);

}

}

return 0;

}

**22. Write a C program to print all odd number between 1 to 100:**

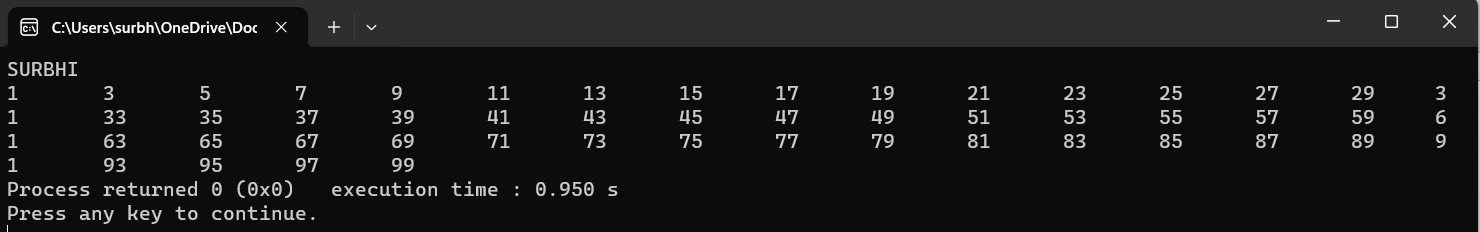
#include<stdio.h>

int main(){

printf("SURBHI\n");

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

if (i % 2 != 0){

 printf("%d\t", i);

}

}

return 0;

}

**23. Write a C program to find sum of all natural numbers between 1 to n:**

#include<stdio.h>

int main(){

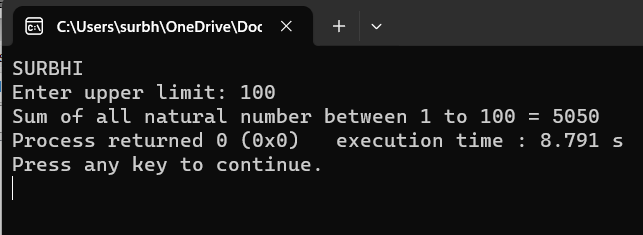
int i, n,sum;

printf("SURBHI\n");

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

if ( sum = 0);{

print(“sum of all natural numbers 1 to n=%d,n,sum);

 }

}

return 0;

}

24.Write a C program to find sum of all even natural numbers between 1 to n:

#include <stdio.h>

int main()

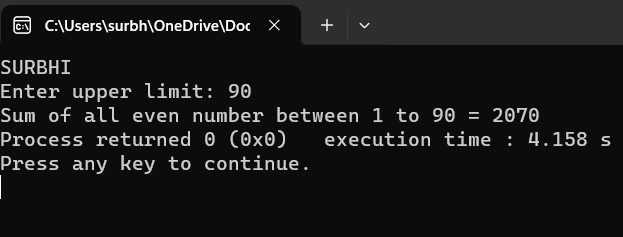
{

int i, n, sum=0;

printf("SURBHI\n");

printf("Enter upper limit: ");

scanf("%d", &n);



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

{

sum += i;

}

printf("Sum of all even number between 1 to n = %d", n, sum);

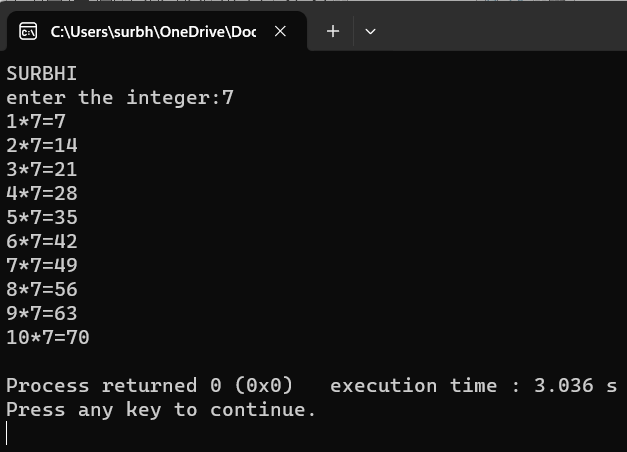
return 0;

}

**25. Write a C program to print multiplication table of any number:**

#include<stdio.h>

int main() {

 int n,i;

printf("SURBHI\n");

printf("enter the integer:");

scanf("%d",&n);

for(int i=1;i<=10;i++) {

printf("%d\*%d=%d\n",i,n,n\*i);

}

return 0;

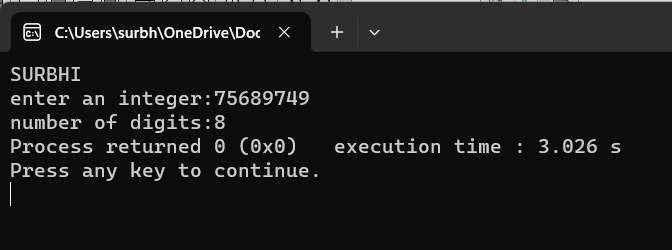
}

**26. Write a C program to count number of digits in a number:**

#include<stdio.h>

int main()

{

 long long n;

int count =0;

printf("SURBHI\n");

printf("enter an integer:");

scanf("%lld",&n);

do{

n/=10;

++count;

}while(n!=0);

printf("number of digits:%d",count);

}

return 0;

}

**27. Write a C program to find first and last digit of a number:**

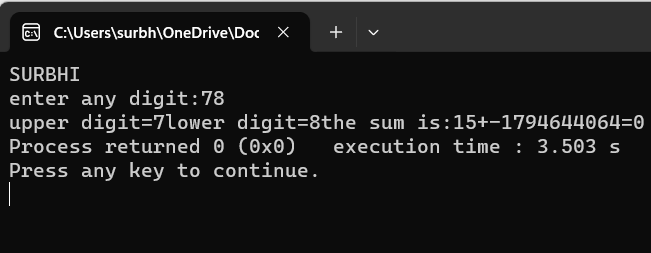
#include<stdio.h>

int main()

{

int n,lowerdigit,upperdigit,sum=0;

printf("SURBHI\n");

 printf("enter any digit:");

scanf("%d",&n);

upperdigit = n/10;

printf("upper digit=%d",upperdigit);

lowerdigit = n%10;

printf("lower digit=%d",lowerdigit);

return 0;

}

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

#**include** <stdio.h>

**int** main()

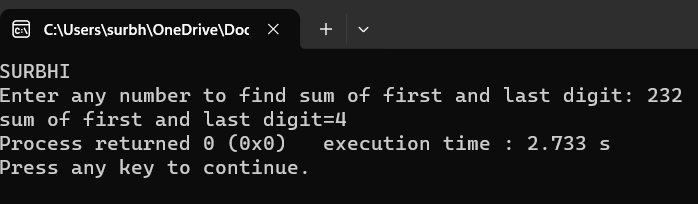
{

**int** num, sum=0, firstDigit, lastDigit;

printf(“SURBHI\n”);

printf("Enter any number to find sum of first and last digit: ");

scanf("%d", &num);

 lastDigit = num % 10;

firstDigit = num;

**while**(num >= 10)

{

num = num / 10;

}

firstDigit = num;

sum = firstDigit + lastDigit;

printf("Sum of first and last digit = %d", sum);

**return** 0;

}

**29. Write a C program to swap first and last digits of a number:**

#include <stdio.h>

#include <math.h>

int main()

{

int n,firstDigit, lastDigit,digits, swappedNum;

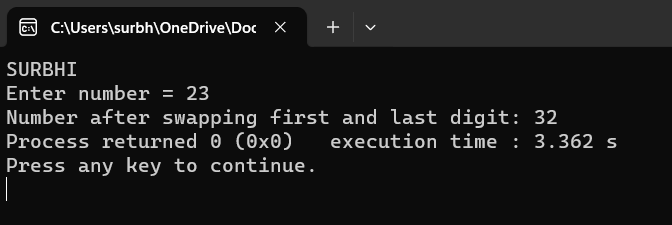
printf("SURBHI\n");

printf("Enter number = ");

scanf("%d", &n);

//Find last digit of a number

lastDigit = n % 10;

 //Find total number of digits - 1

digits = (int)log10(n);

//Find first digit

firstDigit = (int) (n / pow(10, digits));

swappedNum = lastDigit;

swappedNum \*= (int) round(pow(10, digits));

swappedNum += n % ((int)round(pow(10, digits)));

swappedNum -= lastDigit;

swappedNum += firstDigit;

printf("Number after swapping first and last digit: %d", swappedNum);

return 0;

}

**30. Write a C program to calculate sum of first and last digits of a number:**

#include <stdio.h>

int main()

{

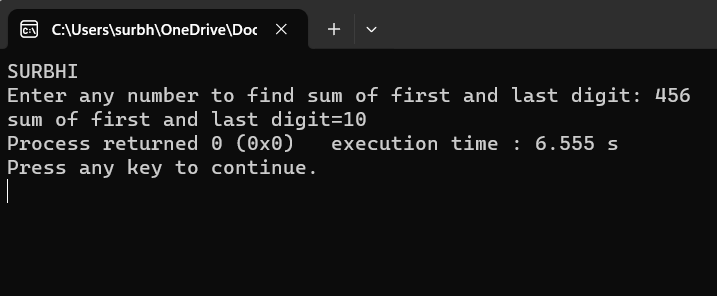
int num, sum=0, firstDigit, lastDigit;

printf("SURBHI\n");

/\* Input a number from user \*/

printf("Enter any number to find sum of first and last digit: ");

scanf("%d", &num);



/\* Find last digit to sum \*/

lastDigit = num % 10;

/\* Copy num to first digit \*/

firstDigit = num;

/\* Find the first digit by dividing num by 10 until first digit is left \*/

while(num >= 10)

{

num = num / 10;

}

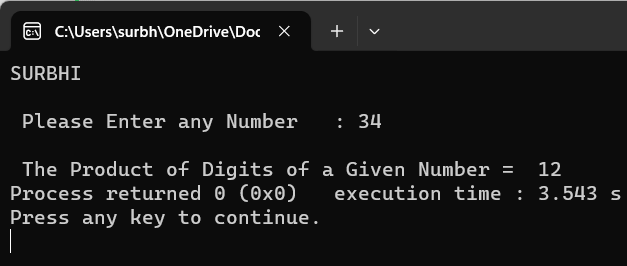
firstDigit = num;

sum=firstDigit + lastDigit;

printf("sum of first and last digit=%d",sum);

return 0;

}

**31. Write a C program to calculate product of digits of a number:**

#include <stdio.h>

int main()

{

int Number, Reminder, Product;

printf(“SURBHI\n”);

printf("\n Please Enter any Number : ");

scanf("%d", & Number);

for(Product = 1; Number > 0; Number = Number / 10)

{

Reminder = Number % 10;

Product = Product \* Reminder;

}

printf(" \n The Product of Digits of a Given Number = %d", Product);

return 0;

}

**32. Write a C program to enter a number and print its reverse:**

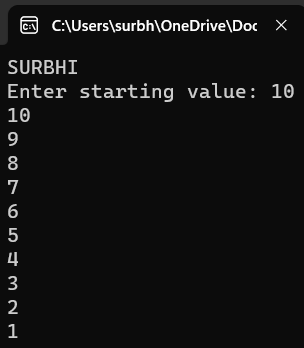
#include <stdio.h>

int main()

{

int i, start;

printf("SURBHI\n");

 printf("Enter starting value: ");

scanf("%d", &start);

for(i=start; i>=1; i--)

{

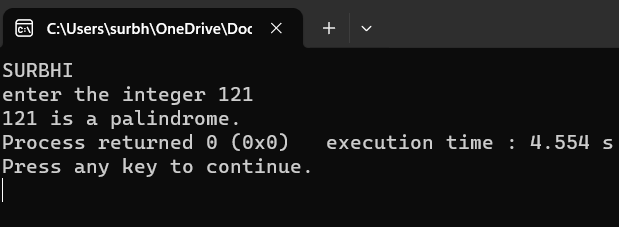
printf("%d\n", i);

}

return 0;

}

**33. Write a C program to check whether a number is palindrome or not:**

#include<stdio.h>

int main()

{

int n,reversed=0,original,remainder;

printf("SURBHI\n");

printf("enter the integer");

scanf("%d",&n);

original=n;

//reversed number is stored in reversed variable

while(n!=0)

{

remainder=n % 10;

reversed = reversed \*10+remainder;

n/=10;

}

//Palindrome if original and reversed are equal

if (original=reversed)

printf("%d is a palindrome.",original);

else

printf("%d is not a palindrome.",original);

return 0;

}

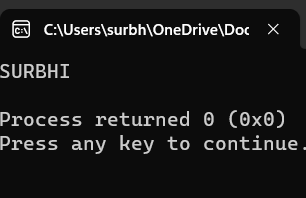
**34. Write a C program to find frequency of each digit in number:**

#include<stdio.h>

int frequencyDigits(int n, int d)

{ int c = 0;

printf("SURBHI\n");

 while (n > 0) {

if (n % 10 == d)

c++;

n = n / 10;

}

return c;

}

int main()

{

int count =0;

int N = 1122322;

int D = 2;

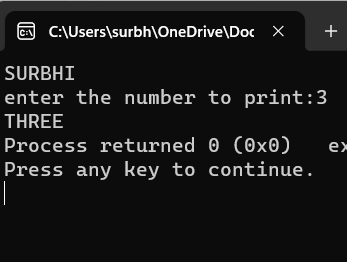
count<<frequencyDigits(N,D);

return 0;

}

**35. Write a C program to enter a number and print it in words:**

#include<stdio.h>

int main()

{

int n , num;

printf(“SURBHI\n”);

printf("enter the number to print:");

scanf("%d",&n);

switch(n%10)

{

case 1:

printf("ONE");

break;

case 2:

printf("TWO");

break;

case 3:

printf("THREE");

break ;

case 4:

printf("FOUR");

break;

default:

printf("ZERO");

}

return 0;

}

**36. Write a C program to print all ASCII character with their values:**

#include <stdio.h>

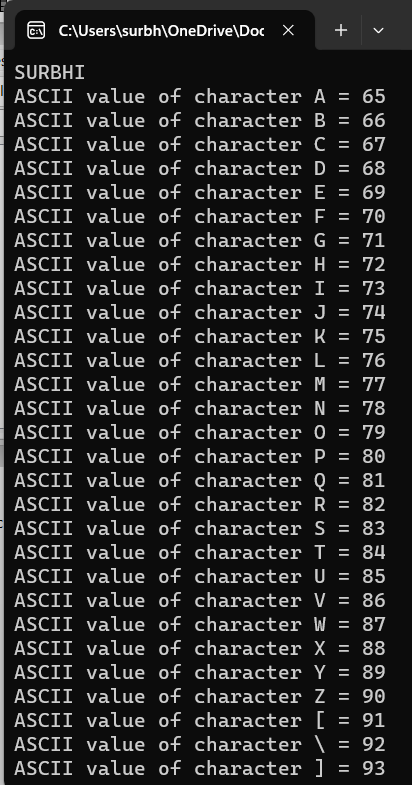
int main()

{

int i;

printf("SURBHI\n");

for(i=65; i<=255; i++)

 {

printf("ASCII value of character %c = %d\n", i, i);

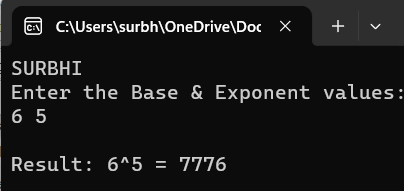
}

return 0;

}

**37. Write a C program to find power of a number using for loop:**

#include <stdio.h>



int main() {

int base, exponent, result = 1;

printf("SURBHI\n");

printf("Enter the Base & Exponent values:\n");

scanf("%d%d", &base, &exponent);

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

result \*= base;

}

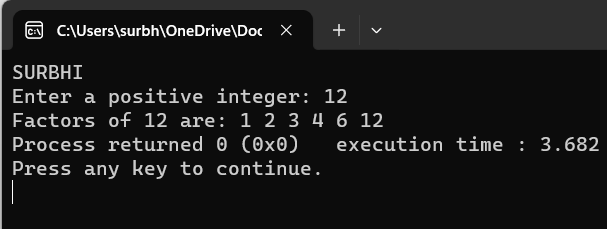
printf("\nResult: %d^%d = %d\n", base, exponent, result);

return 0;

}

**38. Write a C program to find all factors of a number:**

#include <stdio.h>

int main() {

int num, i;

printf("SURBHI\n");

printf("Enter a positive integer: ");

scanf("%d", &num);

printf("Factors of %d are: ", num);

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

if (num % i == 0) {

printf("%d ", i);

}

}

return 0;

}

**39. Write a C program to calculate factorial of a number:**

#include <stdio.h>

unsigned int factorial(unsigned int n)

{

int result = 1, i;

printf("SURBHI\n");

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

result \*= i;

}

return result;

}

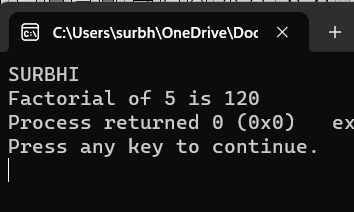
int main()

{

int num = 5;

printf("Factorial of %d is %d", num, factorial(num));

return 0;

}

**40. Write a C program to find HCF (GCD) of two numbers:**

#include <stdio.h>

int main()

{

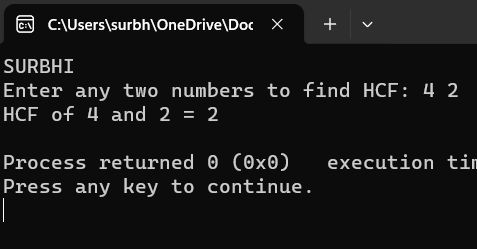
int i, num1, num2, min, hcf=1;

printf("SURBHI\n");

printf("Enter any two numbers to find HCF: ");

scanf("%d%d", &num1, &num2);

min = (num1<num2) ? num1 : num2;

**** for(i=1; i<=min; i++)

{

if(num1%i==0 && num2%i==0)

{

hcf = i;

}

}

printf("HCF of %d and %d = %d\n", num1, num2, hcf);

return 0;

}

**41. Write a C program to find LCM of two numbers:**

#include <stdio.h>

int main()

{

int i, num1, num2, min, lcm=1;

printf("SURBHI\n");

printf("Enter any two numbers to find LCM: ");

scanf("%d%d", &num1, &num2);

min = (num1<num2) ? num1 : num2;

for(i=1; i<=min; i++)

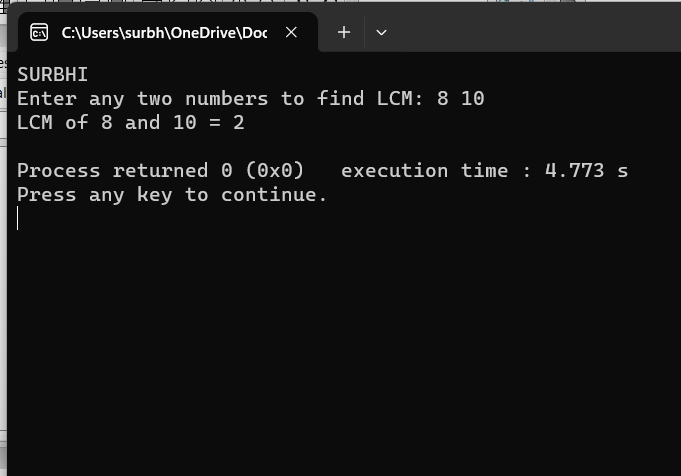
{

if(num1%i==0 && num2%i==0)

{

lcm = i;

}

 }

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

return 0;

}

**42. Write a C program to swap first and last digits of a number:**

#include <stdio.h>

#include <math.h>

int main()

{

int n,firstDigit, lastDigit,digits, swappedNum;

printf("SURBHI\n");

printf("Enter number = ");

scanf("%d", &n);

//Find last digit of a number

lastDigit = n % 10;

//Find total number of digits - 1

digits = (int)log10(n);

//Find first digit

firstDigit = (int) (n / pow(10, digits));

swappedNum = lastDigit;

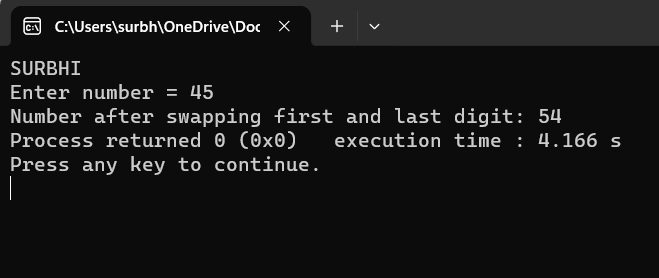
swappedNum \*= (int) round(pow(10, digits));

swappedNum += n % ((int)round(pow(10, digits)));

swappedNum -= lastDigit;

swappedNum += firstDigit;

printf("Number after swapping first and last digit: %d", swappedNum);

 return 0;

}

**43. Write a C program to check whether a number is Prime number or not:**

#include<stdio.h>

int main()

{

int SURBHI,n;

printf( "SURBHI\n");

printf("enter number: ");

scanf("%d",&n);

if (n%2==0)

{

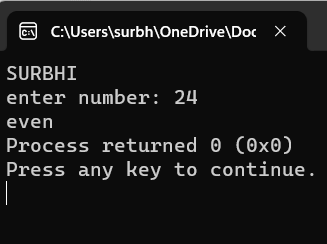
printf("even");

}

else

{

printf("not even");

 }

return 0;

}

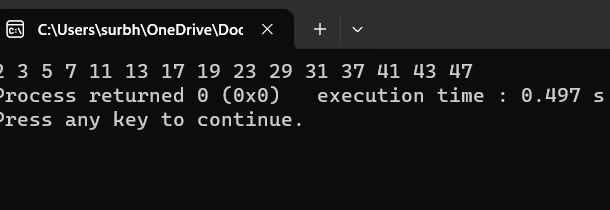
**44. Write a C program to print all Prime numbers between 1 to n:**

#include <stdbool.h>

#include <stdio.h>

bool isPrime(int n)

{

if (n == 1 || n == 0)

return false;

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

if (n % i == 0)

return false;

}

Return true;

}

int main()

{

int N = 50;

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

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

if (isPrime(i)) {

printf("%d ", i);

}

}

return 0;

}

**45. Write a C program to find sum of all prime numbers between 1 to n:**

#include <stdio.h>

int main()

{

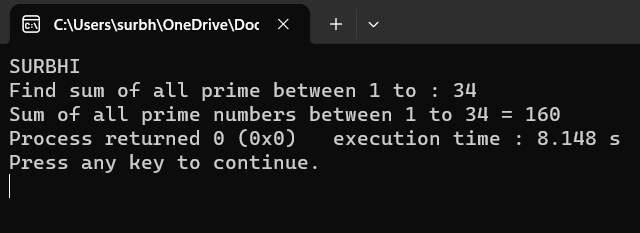
int i, j, end, isPrime, sum=0;

printf("SURBHI\n");

printf("Find sum of all prime between 1 to : ");

scanf("%d", &end);

for(i=2; i<=end; i++)

 {

isPrime = 1;

for(j=2; j<=i/2 ;j++)

{

if(i%j==0)

{

isPrime = 0;

break;

}

}

if(isPrime==1)

{

sum += i;

}

}

printf("Sum of all prime numbers between 1 to %d = %d", end, sum);

return 0;

}

**46. Write a C program to find all prime factors of a number:**

#include <stdio.h>

int main()

{

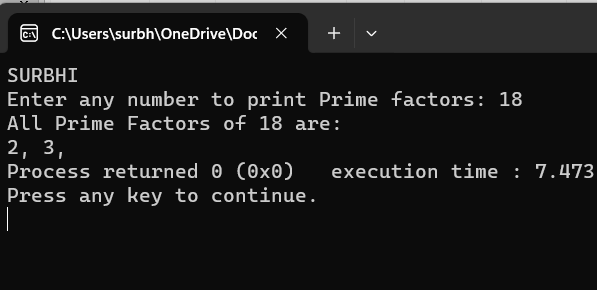
int i, j, num, isPrime;

printf("SURBHI\n");

printf("Enter any number to print Prime factors: ");

scanf("%d", &num);

printf("All Prime Factors of %d are: \n", num);

 for(i=2; i<=num; i++)

{

if(num%i==0)

isPrime = 1;

for(j=2; j<=i/2; j++)

{

if(i%j==0)

{

isPrime = 0;

break;

}

}

if(isPrime==1)

{

printf("%d, ", i);

}

}

return 0

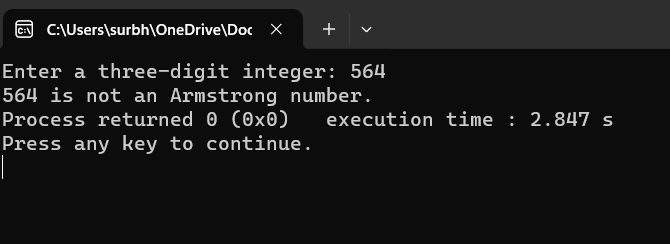
}

**47. Write a C program to check whether a number is Armstrong number or not:**

**ong Number of three digits**

#include <stdio.h>

int main() {

 int num, originalNum, remainder, result = 0;

printf(“SURBHI\n”);

printf("Enter a three-digit integer: ");

scanf("%d", &num);

originalNum = num;

while (originalNum != 0) {

remainder = originalNum % 10;

result += remainder \* remainder \* remainder;

originalNum /= 10;

}

if (result == num)

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

else

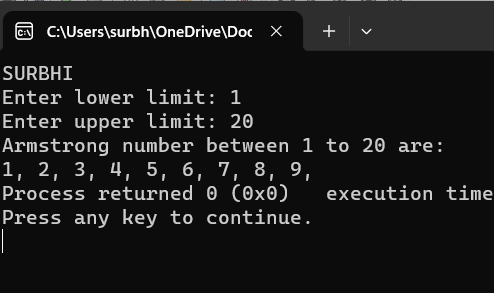
printf("%d is not an Armstrong number.", num);

return 0;

}

**48. Write a C program to print all Armstrong numbers between 1 to n:**

#include <stdio.h>

#include <math.h>

int main()

{ int num, lastDigit, digits, sum, i,start, end;

printf("SURBHI\n");

printf("Enter lower limit: ");

scanf("%d", &start);

printf("Enter upper limit: ");

scanf("%d", &end);

printf("Armstrong number between 1 to %d are: \n", end);

for(i=start; i<=end; i++)

(

sum = 0;

num=i;

digits = (int) log10(num) + 1;

while(num > 0)

{

lastDigit = num % 10;

sum = sum + ceil(pow(lastDigit, digits));

num = num / 10;

}

f(i == sum)

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

}

}

return 0;

}

**49. Write a C program to check whether a number is Perfect number or not:**

#include <stdio.h>

int main()

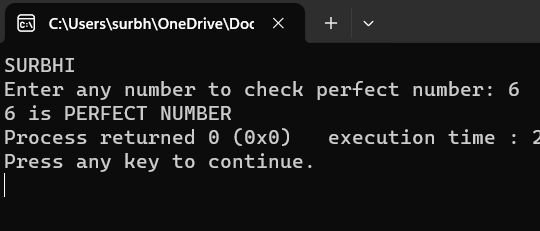
{

int i, num, sum = 0;

printf("SURBHI\n");

printf("Enter any number to check perfect number: ");

scanf("%d", &num);

 for(i = 1; i <= num / 2; i++)

{

if(num%i == 0)

{

sum += i;

}

}

if(sum == num && num > 0)

{

printf("%d is PERFECT NUMBER", num);

}

else

{

printf("%d is NOT PERFECT NUMBER", num);

}

return 0;

}

**50. Write a C program to print all Perfect numbers between 1 to n:**

#include <stdio.h>

int main()

{

int i, j, end, sum;

printf("SURBHI\n");

printf("Enter upper limit: ");

scanf("%d", &end);

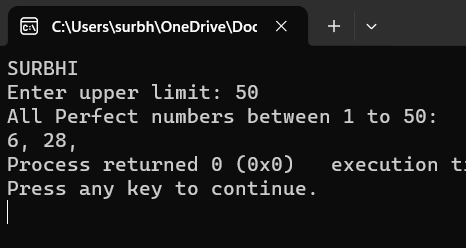
printf("All Perfect numbers between 1 to %d:\n", end);

for(i=1; i<=end; i++)

{

sum = 0;

for(j=1; j<i; j++)

 {

if(i % j == 0)

{

sum += j;

}

}

if(sum == i)

{

printf("%d, ", i);

}

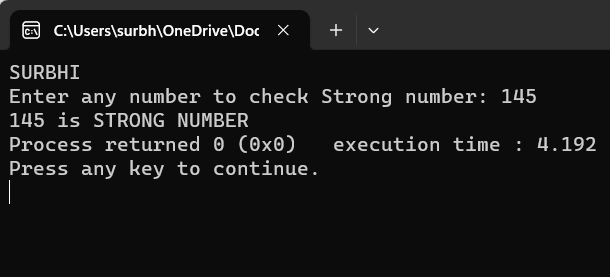
}

return 0;

}

**51.Write a c program whether a number is strong number or not:**

#include <stdio.h>

****int main()

{

int i, originalNum, num, lastDigit, sum;

long fact;

printf("SURBHI\n");

printf("Enter any number to check Strong number: ");

scanf("%d", &num);

originalNum = num;

sum = 0;

while(num > 0)

{

lastDigit = num % 10;

fact = 1;

for(i=1; i<=lastDigit; i++)

{

fact = fact \* i;

}

sum = sum + fact;

num = num / 10;

}

if(sum == originalNum)

{

printf("%d is STRONG NUMBER", originalNum);

}

else

{

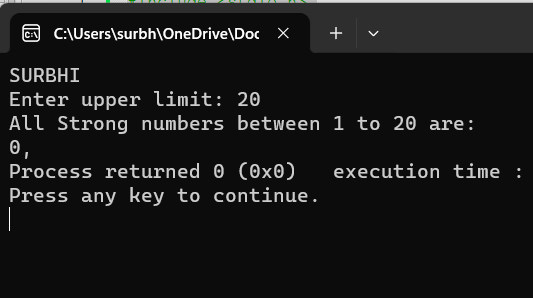
printf("%d is NOT STRONG NUMBER", originalNum);

)

Return 0;

)

**52. Write a C program to print all Strong numbers between 1 to n:**

#include <stdio.h>

int main()

{

int i, j, a, lastDigit, end;

long long fact, sum;

printf("SURBHI\n");

printf("Enter upper limit: ");

scanf("%d", &end);

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

{

a = i;

sum = 0;

while(a > 0)

{

fact = 1ll;

lastDigit =a % 10;

for( j=1; j<=lastDigit; j++)

{

fact = fact \* j;

}

sum += fact;

a /= 10;

}

{

printf("%d, ", i);

}

}

return 0;

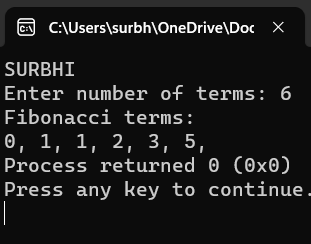
}

**53. Write a C program to print Fibonacci series up to n terms:**

#include <stdio.h>

int main()

{

 int a, b, c, i, terms;

printf("SURBHI\n");

printf("Enter number of terms: ");

scanf("%d", &terms);

a = 0;

b = 1;

c = 0;

printf("Fibonacci terms: \n");

for(i=1; i<=terms; i++)

{

printf("%d, ", c);

a = b;

b = c;

c = a + b;

}

return 0;

}

**54. Write a C program to find one's complement of a binary number:**

#include <stdio.h>

#define SIZE 8

int main()

{

char binary[SIZE + 1], onesComp[SIZE + 1];

int i, error=0;

printf("SURBHI\n");

printf("Enter %d bit binary value: ", SIZE);

gets(binary);

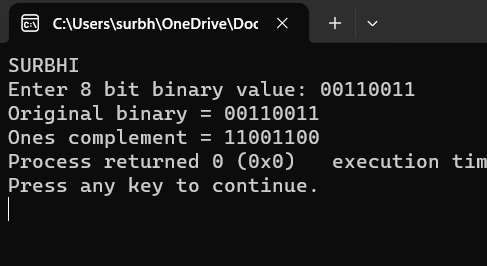
for(i=0; i<SIZE; i++)

{

if(binary[i] == '1')

{

onesComp[i] = '0';

 }

else if(binary[i] == '0')

{

onesComp[i] = '1';

}

else

{

printf("Invalid Input");

error = 1;

break;

}

}

onesComp[SIZE] = '\0';

if(error == 0)

{

printf("Original binary = %s\n", binary);

printf("Ones complement = %s", onesComp);

}

return 0;

}

**55. Write a C program to find two's complement of a binary:**

#include<stdlib.h>

#define SIZE 8

int main(){

int i, carry = 1;

char num[SIZE + 1], one[SIZE + 1], two[SIZE + 1];

printf("SURBHI\n");

printf("Enter the binary number");

gets(num);

for(i = 0; i < SIZE; i++){

if(num[i] == '0'){

one[i] = '1';

}

else if(num[i] == '1'){

one[i] = '0';

}

}

one[SIZE] = '\0';

printf("Ones' complement of binary number %s is %s",num, one);

for(i = SIZE - 1; i >= 0; i--){

if(one[i] == '1' && carry == 1){

two[i] = '0';

}

else if(one[i] == '0' && carry == 1){

two[i] = '1';

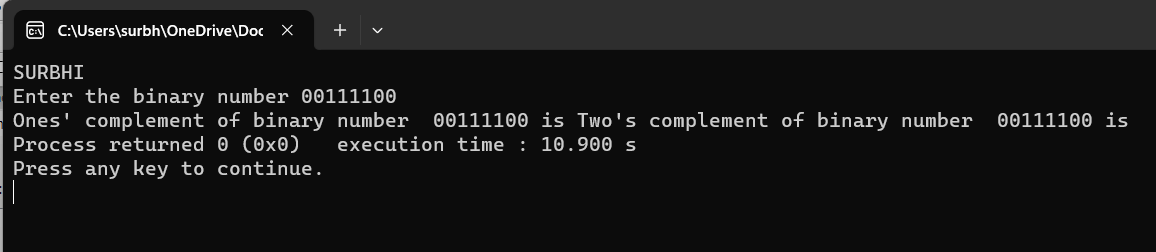
carry = 0;

}

else{

two[i] = one[i];

}

}

**56. Write a C program to convert Binary to Octal number system:**

#include <stdio.h>

int main()

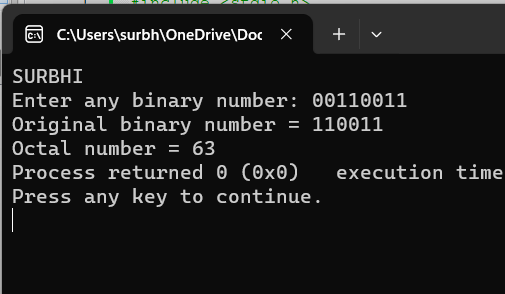
{

int octalConstant[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long binary, octal, tempBinary;

int digit, place, i;

octal = 0;

 place= 1;

printf("SURBHI\n");

printf("Enter any binary number: ");

scanf("%lld", &binary);

tempBinary = binary;

while(tempBinary != 0)

{

digit = tempBinary % 1000;

for(i=0; i<8; i++)

{

if(octalConstant[i] == digit)

{

octal = (i \* place) + octal;

break;

}

}

tempBinary /= 1000;

place \*= 10;

}

printf("Original binary number = %lld\n", binary);

printf("Octal number = %lld", octal);

return 0;

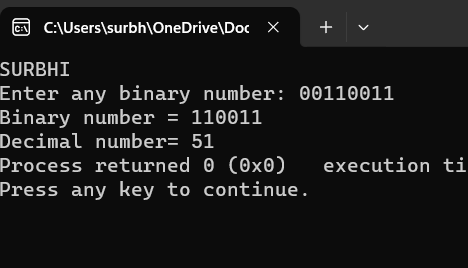
}

**57. Write a C program to convert Binary to Decimal number system:**

#include <stdio.h>

#include <math.h>

#define BASE 2

int main()

{

long long binary, decimal=0, tempBinary;

int N=0;

printf("SURBHI\n");

printf("Enter any binary number: ");

scanf("%lld", &binary);

tempBinary = binary;

while(tempBinary!=0)

{

if(tempBinary % 10 == 1)

{

decimal += pow(BASE, N);

}

N++;

tempBinary /= 10;

}

printf("Binary number = %lld\n", binary);

printf("Decimal number= %lld", decimal);

return 0;

}

**58. Write a C program to convert Binary to Hexadecimal number system:**

int main()

{

int hexConstant[] = {0, 1, 10, 11, 100, 101, 110, 111, 1000,

1001, 1010, 1011, 1100, 1101, 1110, 1111};

long long binary, tempBinary;

char hex[20];

int index, i, digit;

/\* Input binary number from user \*/

printf("SURBHI\n");

printf("Enter binary number: ");

scanf("%lld", &binary);

/\* Copy binary number to temp variable \*/

tempBinary = binary;

index = 0;

/\* Find hexadecimal of binary number \*/

while(tempBinary!=0)

{

/\* Group binary to last four digits \*/

digit = tempBinary % 10000;

/\* Find hexadecimal equivalent of last four digit \*/

for(i=0; i<16; i++)

{

if(hexConstant[i] == digit)

{

if(i<10)

{

/\* 0-9 integer constant \*/

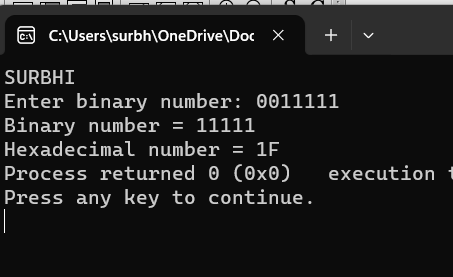
hex[index] = (char)(i + 48);

}

else

{

/\* A-F character constant \*/

 hex[index] = (char)((i-10) + 65);

}

index++;

break;

}

}

/\* Remove the last 4 digits as it is processed \*/

tempBinary /= 10000;

}

hex[index] = '\0';

/\* Reverse the hex digits \*/

strrev(hex);

printf("Binary number = %lld\n", binary);

printf("Hexadecimal number = %s", hex);

return 0;

**59. Write a C program to convert Octal to Binary number system:**

#include <stdio.h>

int main()

{

int OCTALVALUES[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long octal, tempOctal, binary, place;

int rem;

printf("SURBHI\n");

printf("Enter any Octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

binary = 0;

place = 1;

/\* Convert octal to binary \*/

while(tempOctal > 0)

{

/\* Extract the last digit of octal \*/

rem = tempOctal % 10;

/\*

\* Get the binary equivalent of octal digit

\* add it to the binary variable

\*/

binary = (OCTALVALUES[rem] \* place) + binary;

/\* Remove the last octal digit \*/

tempOctal /= 10;

/\* Increase the place value \*/

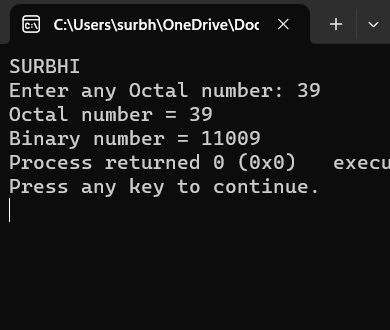
place \*= 1000;

}

printf("Octal number = %lld\n", octal);

printf("Binary number = %lld", binary);

return 0;



}

**60. Write a C program to convert Octal to Decimal number system:**

#include <stdio.h>

#include <math.h>

// function prototype

int convertDecimalToOctal(int decimalNumber);

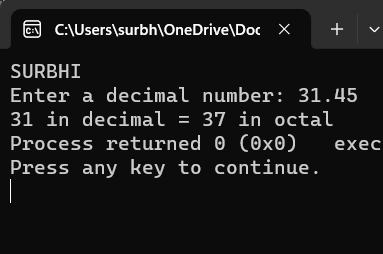
int main() {

int decimalNumber;

printf("Enter a decimal number: ");

scanf("%d", &decimalNumber);

printf("%d in decimal = %d in octal", decimalNumber, convertDecimalToOctal(decimalNumber));

 return 0;

}

// function to convert decimalNumber to octal

int convertDecimalToOctal(int decimalNumber) {

int octalNumber = 0, i = 1;

while (decimalNumber != 0) {

octalNumber += (decimalNumber % 8) \* i;

decimalNumber /= 8;

i \*= 10;

}

return octalNumber;

}

**61. Write a C program to convert Octal to Hexadecimal number system:**

#include <stdio.h>

int main()

{

int OCTALVALUES[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long octal, tempOctal, binary, place;

char hex[65] = "";

int rem;

place = 1;

binary = 0;

printf("SURBHI\n");

printf("Enter any octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

/\*

\* Octal to binary conversion

\*/

while(tempOctal > 0)

{

rem = tempOctal % 10;

binary = (OCTALVALUES[rem] \* place) + binary;

tempOctal /= 10;

place \*= 1000;

}

/\*

\* Binary to hexadecimal conversion

\*/

while(binary > 0)

{

rem = binary % 10000;

switch(rem)

{

case 0:

strcat(hex, "0");

break;

case 1:

strcat(hex, "1");

break;

case 10:

strcat(hex, "2");

break;

case 11:

strcat(hex, "3");

break;

case 100:

strcat(hex, "4");

break;

case 101:

strcat(hex, "5");

break;

case 110:

strcat(hex, "6");

break;

case 111:

strcat(hex, "7");

break;

case 1000:

strcat(hex, "8");

break;

case 1001:

strcat(hex, "9");

break;

case 1010:

strcat(hex, "A");

break;

case 1011:

strcat(hex, "B");

break;

case 1100:

strcat(hex, "C");

break;

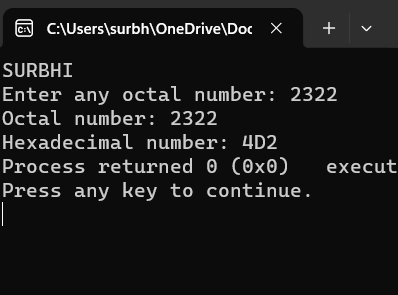
case 1101:

strcat(hex, "D");

break;

case 1110:

strcat(hex, "E");

 break;

case 1111:

strcat(hex, "F");

break;

}

binary /= 10000;

}

strrev(hex);

printf("Octal number: %lld\n", octal);

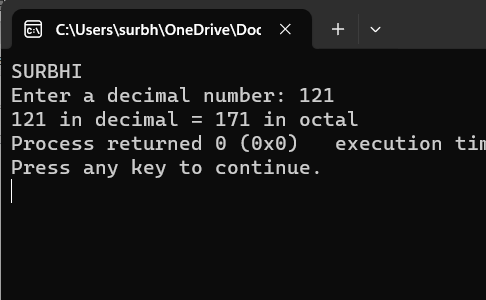
printf("Hexadecimal number: %s", hex);

return 0;

}

**62. Write a C program to convert Decimal to Binary number system:**

#include <stdio.h>

#include <math.h>

int convertDecimalToOctal(int decimalNumber);

int main() {

int decimalNumber;

printf("SURBHI\n");

printf("Enter a decimal number: ");

scanf("%d", &decimalNumber);

printf("%d in decimal = %d in octal", decimalNumber, convertDecimalToOctal(decimalNumber));

return 0;

}

// function to convert decimalNumber to octal

int convertDecimalToOctal(int decimalNumber) {

int octalNumber = 0, i = 1;

while (decimalNumber != 0) {

octalNumber += (decimalNumber % 8) \* i;

decimalNumber /= 8;

i \*= 10;

}

return octalNumber;

}

**63. Write a C program to convert Decimal to Octal number system:**

#include <stdio.h>

#include <math.h>

// function prototype

int convertDecimalToOctal(int decimalNumber);

int main() {

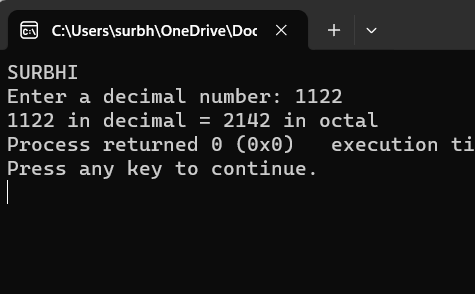
int decimalNumber;

printf("SURBHI\n");

printf("Enter a decimal number: ");

scanf("%d", &decimalNumber);

printf("%d in decimal = %d in octal", decimalNumber, convertDecimalToOctal(decimalNumber));

 return 0;

}

// function to convert decimalNumber to octal

int convertDecimalToOctal(int decimalNumber) {

int octalNumber = 0, i = 1;

while (decimalNumber != 0) {

octalNumber += (decimalNumber % 8) \* i;

decimalNumber /= 8;

i \*= 10;

}

return octalNumber;

)

**64. Write a C program to convert Decimal to Hexadecimal number system:**

#include<stdio.h>

int main()

{

int decimal\_Number = 45;

int i = 1, j, temp;

char hexa\_Number[100];

printf("SURBHI\n");

while (decimal\_Number != 0) {

temp = decimal\_Number % 16;

if (temp < 10)

temp = temp + 48;

else

temp = temp + 55;

hexa\_Number[i++] = temp;

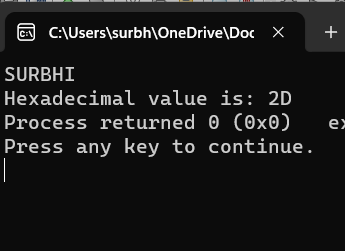
decimal\_Number = decimal\_Number / 16;

}

printf("Hexadecimal value is: ");

for (j = i - 1; j > 0; j--)

printf("%c", hexa\_Number[j]);

 return 0;

**65. Write a C program to convert Hexadecimal to Binary number system:**

include <stdio.h>

#include <string.h>

int main()

{

char hex[17], bin[65] = "";

int i = 0;

/\* Input hexadecimal number from user \*/

printf("Enter any hexadecimal number: ");

gets(hex);

/\* Extract first digit and find binary of each hex digit \*/

for(i=0; hex[i]!='\0'; i++)

{

switch(hex[i])

{

case '0':

strcat(bin, "0000");

break;

case '1':

strcat(bin, "0001");

break;

case '2':

strcat(bin, "0010");

break;

case '3':

strcat(bin, "0011");

break;

case '4':

strcat(bin, "0100");

break;

case '5':

strcat(bin, "0101");

break;

case '6':

strcat(bin, "0110");

break;

case '7':

strcat(bin, "0111");

break;

case '8':

strcat(bin, "1000");

break;

case '9':

strcat(bin, "1001");

break;

case 'a':

case 'A':

strcat(bin, "1010");

break;

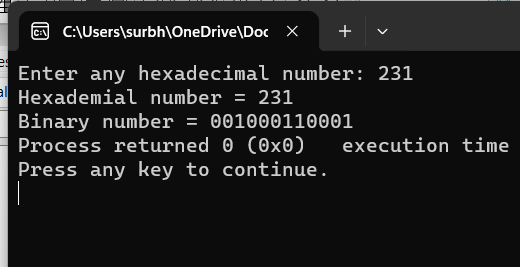
case 'b':

case 'B':

strcat(bin, "1011");

break;

case 'c':

 case 'C':

strcat(bin, "1100");

break;

case 'd':

case 'D':

strcat(bin, "1101");

break;

case 'e':

case 'E':

strcat(bin, "1110");

break;

case 'f':

case 'F':

strcat(bin, "1111");

break;

default:

printf("Invalid hexadecimal input.");

}

}

printf("Hexademial number = %s\n", hex);

printf("Binary number = %s", bin);

return 0;

}

**66. Write a C program to convert Hexadecimal to Octal number system:**

include <stdio.h>

#include <string.h>

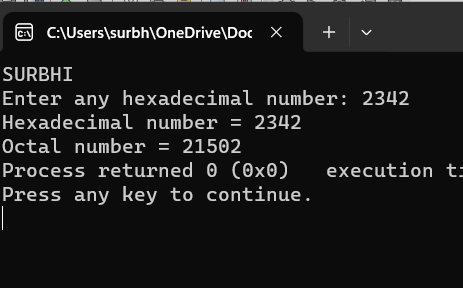
#include <math.h>

//To convert hex to binary first

long long int hexa\_binary(char hex[]) {

long long int binary, place;

int i = 0, rem, val;

 int n = strlen(hex);

binary = 0ll;

place = 0ll;

for (i = 0; hex[i] != '\0'; i++) {

binary = binary \* place;

switch (hex[i]) {

case '0':

binary += 0;

break;

case '1':

binary += 1;

break;

case '2':

binary += 10;

break;

case '3':

binary += 11;

break;

case '4':

binary += 100;

break;

case '5':

binary += 101;

break;

case '6':

binary += 110;

break;

case '7':

binary += 111;

break;

case '8':

binary += 1000;

break;

case '9':

binary += 1001;

break;

case 'a':

case 'A':

binary += 1010;

break;

case 'b':

case 'B':

binary += 1011;

break;

case 'c':

case 'C':

binary += 1100;

break;

case 'd':

case 'D':

binary += 1101;

break;

case 'e':

case 'E':

binary += 1110;

break;

case 'f':

case 'F':

binary += 1111;

break;

default:

printf("Invalid hexadecimal input.");

}

place = 10000;

}

return binary;

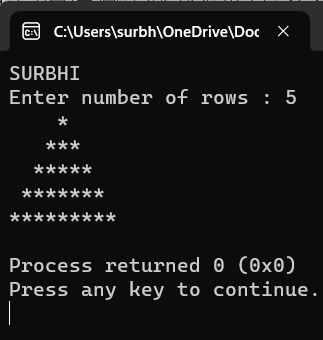
}

***PATTERN EXERCISE:***

1. **Star pattern programs - Write a C program to print the given star patterns:**
2. **.Pyramid Star Pattern:**

#include<stdio.h>

int main()

****{

int i, j, rows;

printf("SURBHI\n");

printf("Enter number of rows : ");

scanf("%d", &rows);

/\* Iterate through rows \*/

for(i=1; i<=rows; i++)

{

/\* Print leading spaces \*/

for(j=i; j<rows; j++)

{

printf(" ");

}

/\* Print star \*/

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

{

printf("\*");

/\* Move to next line \*/

printf("\n");

}

return 0;

}

1. **Hollow Pyramid:**

#include <stdio.h>

int main()

{

int i, j, rows;

printf("SURBHI\n");

printf("Enter number of rows : ");

scanf("%d", &rows);

for(i=1; i<=rows; i++)

{

/\* Print trailing spaces \*/

for(j=i; j<rows; j++)

{

printf(" ");

}

/\* Print hollow pyramid \*/

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

{

/\*

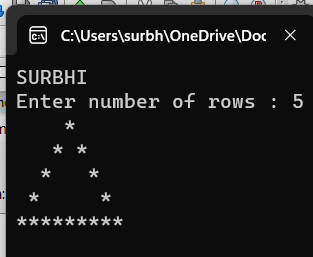
\* Print star for last row (i==rows),

\* first column(j==1) and for

\* last column (j==(2\*i-1)).

\*/

if(i==rows || j==1 || j==(2\*i-1))

 {

printf("\*");

}

else

{

printf(" ");

}

}

/\* Move to next line \*/

printf("\n");

}

return 0;

}

**C). Inverted Star Pattern:**

#include <stdio.h>

int main()

{

int i, j, rows;

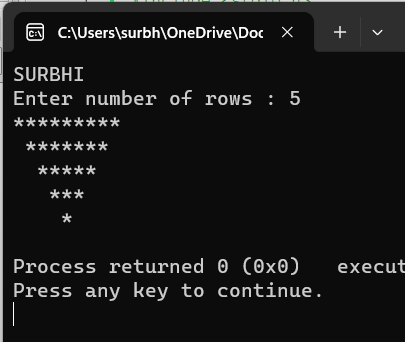
printf("SURBHI\n");

printf("Enter number of rows : ");

scanf("%d", &rows);

for(i=1; i<=rows; i++)

{

 /\* Print leading spaces \*/

for(j=1; j<i; j++)

{

printf(" ");

}

/\* Print stars \*/

for(j=1; j<=(rows\*2 -(2\*i-1)); j++)

{

printf("\*");

}

printf("\n");

}

return 0;

}

**D) Hollow Inverted Star Pattern:**

#include <stdio.h>

int main()

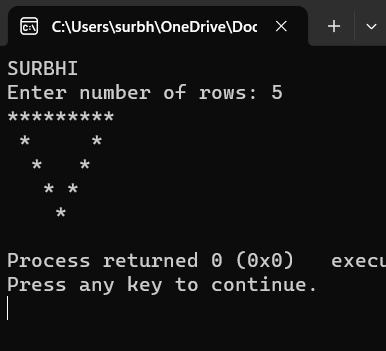
{

int i, j, rows;

printf("SURBHI\n");

printf("Enter number of rows: ");

scanf("%d", &rows);



/\* Iterate through rows \*/

for(i=1; i<=rows; i++)

{

/\* Print leading spaces \*/

for(j=1; j<i; j++)

{

printf(" ");

}

/\* Print hollow pyramid \*/

for(j=1; j<=(rows\*2 - (2\*i-1)); j++)

if(i==1 || j==1 || j==(rows\*2 - (2\*i - 1)))

{

printf("\*");

}

else

{

printf(" ");

}

}

printf("\n");

}

return 0;

}

**E). Half Diamond Star Pattern:**

#include<stdio.h>

int main()

{

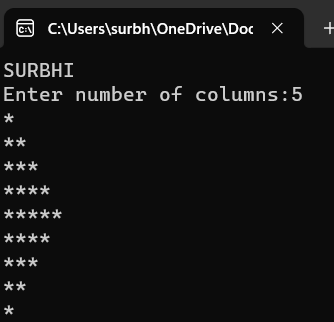
int i, j, N, columns;

printf("SURBHI\n");

printf("Enter number of columns:");

scanf("%d",&N);

columns=1;

for(i=1;i<N\*2;i++)

{

for(j=1; j<=columns; j++)

{

printf("\*");

}

if(i < N)

{

columns++;

}

else

{

columns--;

}

printf("\n");

}

return 0;

}

**F). Mirrored Half Diamond Pattern:**

#include <stdio.h>

int main()

{

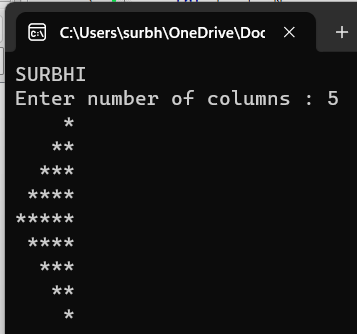
int i, j, N;

int star, spaces;

printf("SURBHI\n");

printf("Enter number of columns : ");

scanf("%d", &N);

 spaces = N-1;

star = 1;

for(i=1; i<N\*2; i++)

{

for(j=1; j<=spaces; j++)

printf(" ");

for(j=1; j<=star; j++)

printf("\*");

printf("\n");

if(i < N)

{

star++;

spaces--;

}

else

{

star--;

spaces++;

}

}

return 0;

}

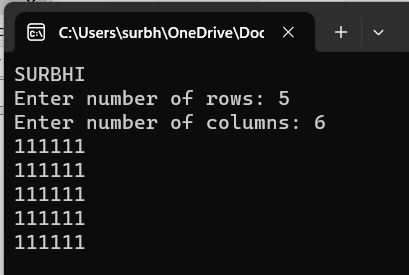
1. **Number pattern programs - Write a C program to print the given number patterns:**

**A).Square Number Pattern:**

#include <stdio.h>

int main()

{

 int rows, cols, i, j;

printf("SURBHI\n");

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

printf("1");

}

printf("\n");

}

return 0;

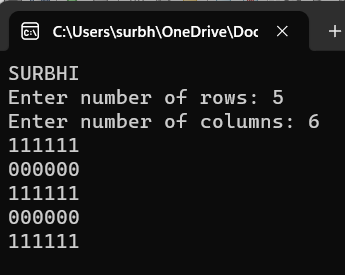
}

**B). Number Pattern 1:**

#include <stdio.h>

int main()

{

 int rows, cols, i, j;

printf("SURBHI\n");

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(i%2 == 1)

{

printf("1");

else

{

printf("0");

}

}

printf("\n");

}

return 0;

}

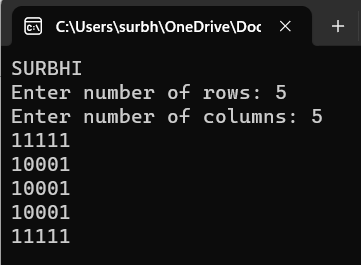
**C). Number Pattern 3:**

#include <stdio.h>

int main()

{

int rows, cols, i, j;

 printf("SURBHI\n");

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(i==1 || i==rows || j==1 || j==cols)

{

printf("1");

}

else

{

printf("0");

}

printf("\n");

}

return 0;

}

**D). Number Pattern 4:**

#include <stdio.h>

int main()

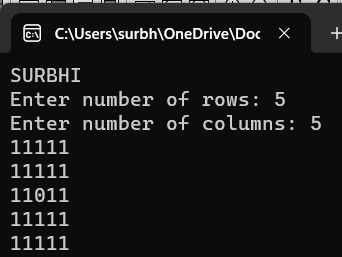
{

int rows, cols, i, j;

int centerRow, centerCol;

printf("SURBHI\n");

printf("Enter number of rows: ");

 scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);

centerRow = (rows + 1) / 2;

centerCol = (cols + 1) / 2;

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(centerCol == j && centerRow == i)

{

printf("0");

}

else if(cols%2 == 0 && centerCol+1 == j)

{

if(centerRow == i || (rows%2 == 0 && centerRow+1 == i))

printf("0");

else

printf("1");

}

else if(rows%2 == 0 && centerRow+1 == i)

{

if(centerCol == j || (cols%2 == 0 && centerCol+1 == j))

printf("0");

else

printf("1");

}

else

{

printf("1");

}

}

printf("\n");

}

return 0;

}

**E). Pattern Number 5:**

#include <stdio.h>

int main()

{

int rows, cols, i, j, k;

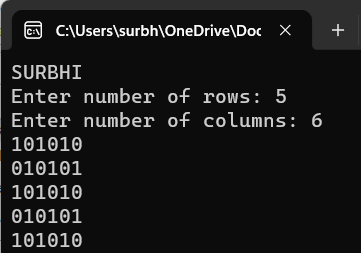
printf("SURBHI\n");

printf("Enter number of rows: ");

scanf("%d", &rows);

printf("Enter number of columns: ");

scanf("%d", &cols);



k = 1;

for(i=1; i<=rows; i++)

{

for(j=1; j<=cols; j++)

{

if(k == 1)

{

printf("1");

}

else

{

printf("0");

}

k \*= -1;

}

if(cols % 2 == 0)

{

k \*= -1;

}

printf("\n");

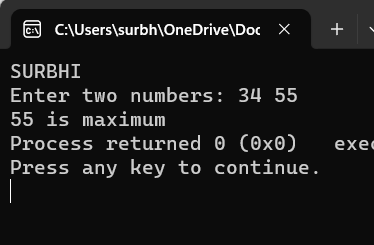
}

return 0;

}

**If…Else Exercises:**

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

#include <stdio.h>

int main()

{

int num1, num2;

printf("SURBHI\n");

printf("Enter two numbers: ");

scanf("%d%d", &num1, &num2);

if(num1 > num2)

{

printf("%d is maximum", num1);

}

if(num2 > num1)

{

printf("%d is maximum", num2);

}

if(num1 == num2)

{

printf("Both are equal");

}

return 0;

}

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

#include <stdio.h>

int main()

{

int num1, num2, num3, max;

printf("SURBHI\n");

printf("Enter three numbers: ");

scanf("%d%d%d", &num1, &num2, &num3);

if(num1 > num2)

{

if(num1 > num3)

{

max = num1;

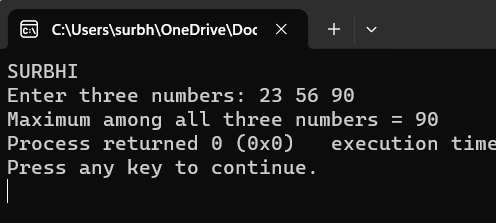
}

else

{

max = num3;

}

 }

else

{

if(num2 > num3)

{

max = num2;

}

else

{

max = num3;

}

}

printf("Maximum among all three numbers = %d", max);

return 0;

}

1. **Write a C program to check whether a number is negative, positive or zero:**

#include <stdio.h>

int main()

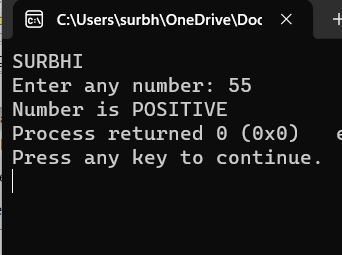
{

int num;

printf("SURBHI\n");

printf("Enter any number: ");

scanf("%d", &num);

 if(num > 0)

{

printf("Number is POSITIVE");

}

if(num < 0)

{

printf("Number is NEGATIVE");

}

if(num == 0)

{

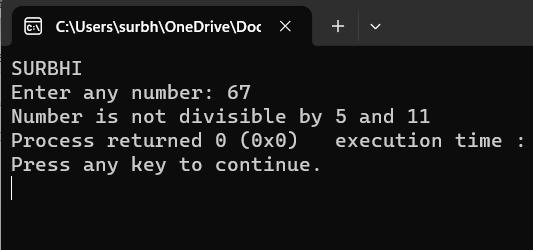
printf("Number is ZERO");

}

return 0;

}

1. **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("SURBHI\n");

printf("Enter any number: ");

scanf("%d", &num);

if((num % 5 == 0) && (num % 11 == 0))

{

printf("Number is divisible by 5 and 11");

}

else

{

printf("Number is not divisible by 5 and 11");

}

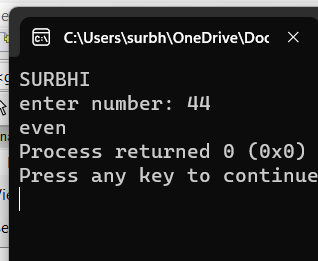
return 0;

}

1. **Write a C program to check whether a number is even or odd:**

#include<stdio.h>

int main()

{

int SURBHI,n;

printf( "SURBHI\n");

printf("enter number: ");

scanf("%d",&n);

if (n%2==0)

{

printf("even");

}

else

{

printf("not even");

}

return 0;

}

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

#include <stdio.h>

int main()

{

int year;

printf("SURBHI\n");

printf("Enter year : ");

scanf("%d", &year);

if(((year % 4 == 0) && (year % 100 !=0)) || (year % 400==0))

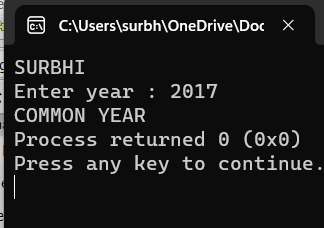
{

printf("LEAP YEAR");

}

else

{

 printf("COMMON YEAR");

}

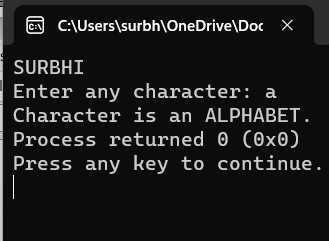
return 0;

}

1. **Write a C program to check whether a character is alphabet or not:**

#include <stdio.h>

int main()

{

char ch;

printf("SURBHI\n");

printf("Enter any character: ");

scanf("%c", &ch);

if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))

{

printf("Character is an ALPHABET.");

}

else

{

printf("Character is NOT ALPHABET.");

}

return 0;

}

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

#include <stdio.h>

int main()

{

char ch;

printf("SURBHI\n");

printf("Enter any alphabet: ");

scanf("%c", &ch);

switch(ch)

{

case 'a':

printf("Vowel");

break;

case 'e':

printf("Vowel");

break;

case 'i':

printf("Vowel");

break;

case 'o':

printf("Vowel");

break;

case 'u':

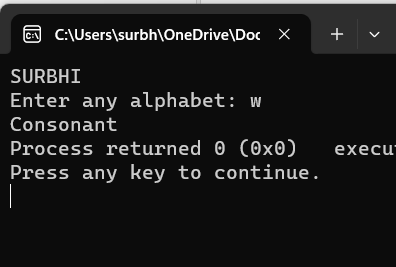
printf("Vowel");

break;

case 'A':

printf("Vowel");

break;

 case 'E':

printf("Vowel");

break;

case 'I':

printf("Vowel");

break;

case 'O':

printf("Vowel");

break;

case 'U':

printf("Vowel");

break;

default:

printf("Consonant");

}

return 0;

}

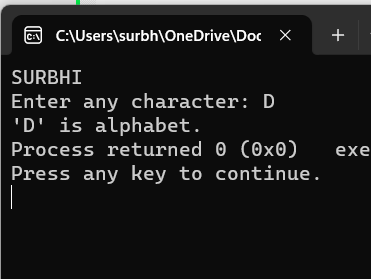
1. **Write a C program to input any character and check whether it is alphabet, digit or special character:**

#include <stdio.h>

int main()

{

char ch;

 printf("SURBHI\n");

printf("Enter any character: ");

scanf("%c", &ch);

if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))

{

printf("'%c' is alphabet.", ch);

}

else if(ch >= '0' && ch <= '9')

{

printf("'%c' is digit.", ch);

}

else

{

printf("'%c' is special character.", ch);

}

return 0;

}

1. **Write a C program to check whether a character is uppercase or lowercase alphabet:**

#include<stdio.h>

int main()

{

char c;

printf("SURBHI\n");

printf("enter any character n:");

scanf("%c",&c);

if(c>'63'&& c<='31')

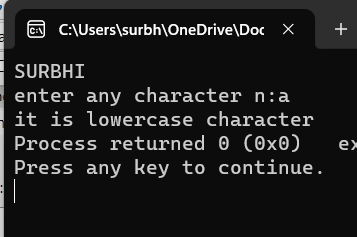
{

printf("it is uppercase character");

}

else if (c>='97' && c>='90');

{

 printf("it is lowercase character");

}

return 0;

}

1. **Write a C program to input week number and print week day:**

#include <stdio.h>

int main()

{

int week;

printf("SURBHI\n");

printf("Enter week number (1-7): ");

scanf("%d", &week);

if(week == 1)

{

printf("Monday");

}

else if(week == 2)

{

printf("Tuesday");

}

else if(week == 3)

{

printf("Wednesday");

}

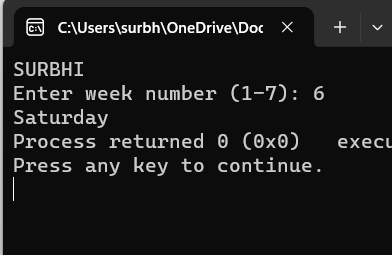
else if(week == 4)

{

printf("Thursday");

}

else if(week == 5)

 {

printf("Friday");

}

else if(week == 6)

{

printf("Saturday");

}

else if(week == 7)

{

printf("Sunday");

}

else

{

printf("Invalid Input! Please enter week number between 1-7.");

}

return 0;

}

1. **Write a C program to input month number and print number of days in that month:**

#include <stdio.h>

int main()

{

int month;

printf("SURBHI\n");

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

scanf("%d", &month);

if(month == 1)

{

printf("31 days");

}

else if(month == 2)

{

printf("28 or 29 days");

}

else if(month == 3)

{

printf("31 days");

}

else if(month == 4)

{

printf("30 days");

}

else if(month == 5)

{

printf("31 days");

}

else if(month == 6)

{

printf("30 days");

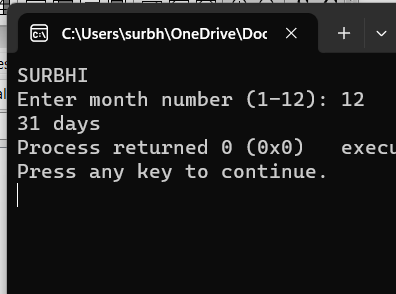
}

else if(month == 7)

{

printf("31 days");

}

 else if(month == 8)

{

printf("31 days");

}

else if(month == 9)

{

printf("30 days");

}

else if(month == 10)

{

printf("31 days");

}

else if(month == 11)

{

printf("30 days");

}

else if(month == 12)

{

printf("31 days");

}

else

{

printf("Invalid input! Please enter month number between (1-12).");

}

return 0;

}

1. **Write a C program to count total number of notes in given amount:**

#include <stdio.h>

int main()

{

int amount;

int note500, note100, note50, note20, note10, note5, note2, note1;

note500 = note100 = note50 = note20 = note10 = note5 = note2 = note1 = 0;

printf("SURBHI\n");

printf("Enter amount: ");

scanf("%d", &amount);

if(amount >= 500)

{

note500 = amount/500;

amount -= note500 \* 500;

}

if(amount >= 100)

{

note100 = amount/100;

amount -= note100 \* 100;

}

if(amount >= 50)

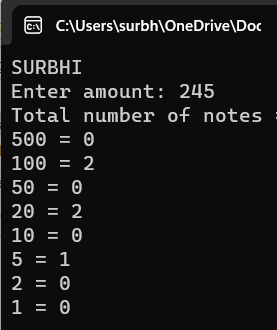
{

note50 = amount/50;

amount -= note50 \* 50;

}

if(amount >= 20)

 {

note20 = amount/20;

amount -= note20 \* 20;

}

if(amount >= 10)

{

note10 = amount/10;

amount -= note10 \* 10;

}

if(amount >= 5)

{

note5 = amount/5;

amount -= note5 \* 5;

}

if(amount >= 2)

{

note2 = amount /2;

amount -= note2 \* 2;

}

if(amount >= 1)

{

note1 = amount;

}

printf("Total number of notes = \n");

printf("500 = %d\n", note500);

printf("100 = %d\n", note100);

printf("50 = %d\n", note50);

printf("20 = %d\n", note20);

printf("10 = %d\n", note10);

printf("5 = %d\n", note5);

printf("2 = %d\n", note2);

printf("1 = %d\n", note1);

return 0;

}

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

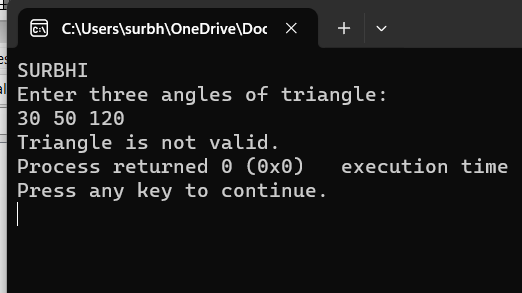
#include <stdio.h>

int main()

{

int angle1, angle2, angle3, sum;

printf("SURBHI\n");

 printf("Enter three angles of triangle: \n");

scanf("%d%d%d", &angle1, &angle2, &angle3);

sum = angle1 + angle2 + angle3;

if(sum == 180 && angle1 > 0 && angle2 > 0 && angle3 > 0)

{

printf("Triangle is valid.");

}

else

{

printf("Triangle is not valid.");

}

return 0;

}

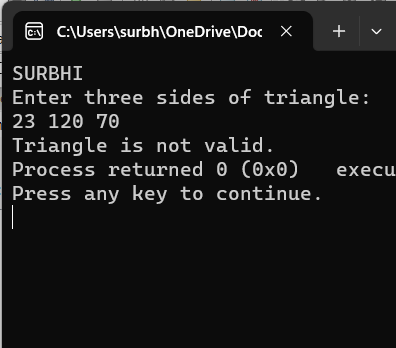
**17. Write a C program to input all sides of a triangle and check whether triangle is valid or not:**

#include <stdio.h>

int main()

{

int side1, side2, side3;

 printf("SURBHI\n");

printf("Enter three sides of triangle: \n");

scanf("%d%d%d", &side1, &side2, &side3);

if((side1 + side2) > side3)

{

if((side2 + side3) > side1)

{

if((side1 + side3) > side2)

{

printf("Triangle is valid.");

}

else

{

printf("Triangle is not valid.");

}

}

else

{

printf("Triangle is not valid.");

}

}

else

{

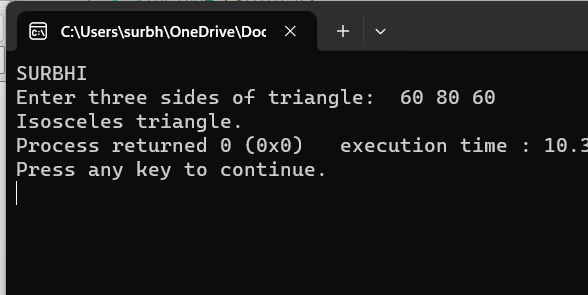
printf("Triangle is not valid.");

}

return 0;

}

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

#include <stdio.h>

int main()

{

int side1, side2, side3;

printf("SURBHI\n");

printf("Enter three sides of triangle: ");

scanf("%d%d%d", &side1, &side2, &side3);

if(side1==side2 && side2==side3)

{

/\* If all sides are equal \*/

printf("Equilateral triangle.");

}

else if(side1==side2 || side1==side3 || side2==side3)

{

/\* If any two sides are equal \*/

printf("Isosceles triangle.");

}

else

{

/\* If none sides are equal \*/

printf("Scalene triangle.");

}

return 0;

}

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

#include <stdio.h>

#include <math.h>

int main()

{

float a, b, c;

float root1, root2, imaginary;

float discriminant;

printf("SURBHI\n");

printf("Enter values of a, b, c of quadratic equation (aX^2 + bX + c): ");

scanf("%f%f%f", &a, &b, &c);

discriminant = (b \* b) - (4 \* a \* c);

if(discriminant > 0)

{

root1 = (-b + sqrt(discriminant)) / (2\*a);

root2 = (-b - sqrt(discriminant)) / (2\*a);

printf("Two distinct and real roots exists: %.2f and %.2f", root1, root2);

}

else if(discriminant == 0)

{

root1 = root2 = -b / (2 \* a);

printf("Two equal and real roots exists: %.2f and %.2f", root1, root2);

}

else if(discriminant < 0)

{

root1 = root2 = -b / (2 \* a);

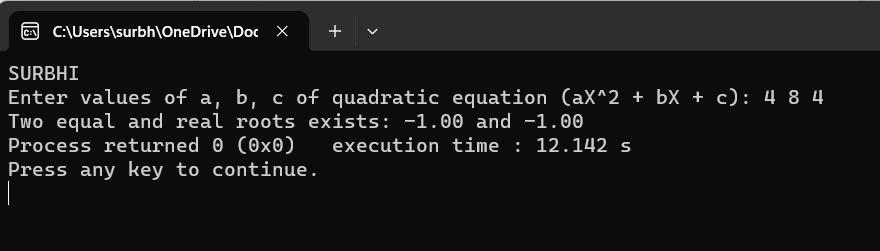
imaginary = sqrt(-discriminant) / (2 \* a);

printf("Two distinct complex roots exists: %.2f + i%.2f and %.2f - i%.2f",

root1, imaginary, root2, imaginary);

}

return 0;

}

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

int main()

{

int cp,sp, amt;

printf("SURBHI\n");

printf("Enter cost price: ");

scanf("%d", &cp);

printf("Enter selling price: ");

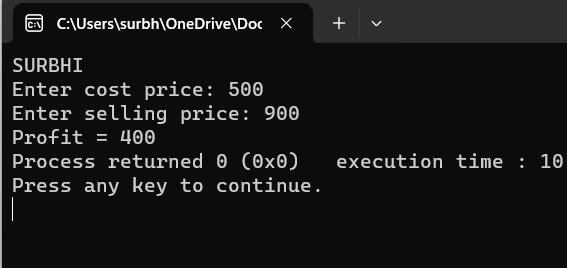
scanf("%d", &sp);

if(sp > cp)

{

/\* Calculate Profit \*/

amt = sp - cp;

 printf("Profit = %d", amt);

}

else if(cp > sp)

{

/\* Calculate Loss \*/

amt = cp - sp;

printf("Loss = %d", amt);

}

else

{

/\* Neither profit nor loss \*/

printf("No Profit No Loss.");

}

return 0;

}

**21. Write a C program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade according to following: Percentage >= 90% : Grade A Percentage >= 80% : Grade B Percentage >= 70% : Grade C Percentage >= 60% : Grade D Percentage >= 40% : Grade E Percentage < 40% : Grade F:**

#include <stdio.h>

int main()

{

int phy, chem, bio, math, comp;

float per;

printf("SURBHI\n");

printf("Enter five subjects marks: ");

scanf("%d%d%d%d%d", &phy, &chem, &bio, &math, &comp);

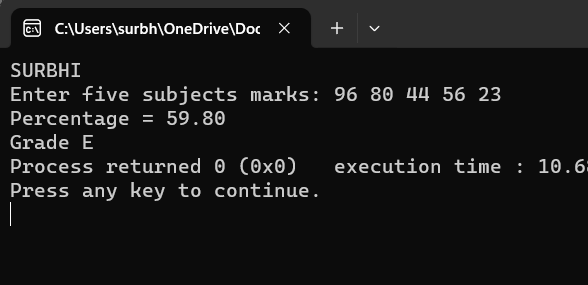
/\* Calculate percentage \*/

per = (phy + chem + bio + math + comp) / 5.0;

printf("Percentage = %.2f\n", per);

if(per >= 90)

{

 printf("Grade A");

}

else if(per >= 80)

{

printf("Grade B");

}

else if(per >= 70)

{

printf("Grade C");

}

else if(per >= 60)

{

printf("Grade D");

}

else if(per >= 40)

{

printf("Grade E");

}

else

{

printf("Grade F");

}

return 0;

}

**22. Write a C program to input basic salary of an employee and calculate its Gross salary according to following: Basic Salary <= 10000 : HRA = 20%, DA = 80% Basic Salary <= 20000 : HRA = 25%, DA = 90% Basic Salary > 20000 : HRA = 30%, DA**

#include <stdio.h>

int main()

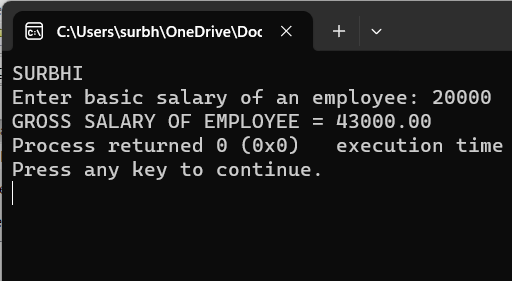
{

float basic, gross, da, hra;

printf("SURBHI\n");

printf("Enter basic salary of an employee: ");

scanf("%f", &basic);



if(basic <= 10000)

{

da = basic \* 0.8;

hra = basic \* 0.2;

}

else if(basic <= 20000)

{

da = basic \* 0.9;

hra = basic \* 0.25;

}

else

{

da = basic \* 0.95;

hra = basic \* 0.3;

}

/\* Calculate gross salary \*/

gross = basic + hra + da;

printf("GROSS SALARY OF EMPLOYEE = %.2f", gross);

return 0;

}

**23. 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()

{

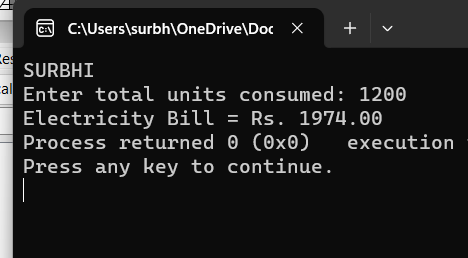
int unit;

float amt, total\_amt, sur\_charge;

printf("SURBHI\n");

printf("Enter total units consumed: ");

scanf("%d", &unit);



if(unit <= 50)

{

amt = unit \* 0.50;

}

else if(unit <= 150)

{

amt = 25 + ((unit-50) \* 0.75);

}

else if(unit <= 250)

{

amt = 100 + ((unit-150) \* 1.20);

}

else

{

amt = 220 + ((unit-250) \* 1.50);

}

/\*

\* Calculate total electricity bill

\* after adding surcharge

\*/

sur\_charge = amt \* 0.20;

total\_amt = amt + sur\_charge;

printf("Electricity Bill = Rs. %.2f", total\_amt);

return 0;

}

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

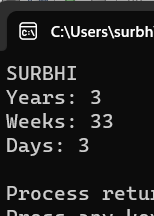
#include <stdio.h>

int main()

{

int days, years, weeks;

days = 1329;



years = days/365;

weeks = (days % 365)/7;

days = days- ((years\*365) + (weeks\*7));

printf("SURBHI\n");

printf("Years: %d\n", years);

printf("Weeks: %d\n", weeks);

printf("Days: %d \n", days);

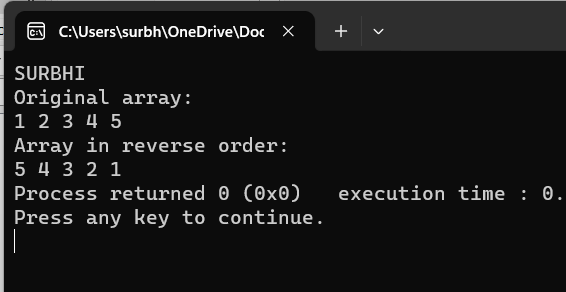
return 0;

}

**QUESTIONS RELATED TO ARRAY:**

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

#include <stdio.h>



int main()

{

//Initialize array

int arr[] = {1, 2, 3, 4, 5};

int length = sizeof(arr)/sizeof(arr[0]);

printf("SURBHI\n");

printf("Original array: \n");

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

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

}

printf("\n");

printf("Array in reverse order: \n");

//Loop through the array in reverse order

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

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

}

return 0;

}

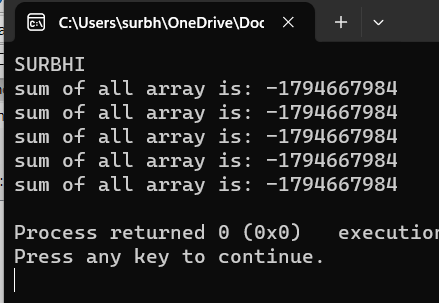
1. **Write a program in C to find the sum of all elements of the array:**

#include<stdio.h>

int main()

{

int i;

 int arr[]={1,2,3,4,5};

int sum=0;

int length=sizeof(arr)/sizeof(arr[0]);

printf("SURBHI\n");

for(i=0;i<length;i++)

{

sum = sum + arr[i];

printf("sum of all array is: %d\n");

}

return 0;

}

1. **Write a program in C to copy the elements of one array into another array:**

#include<stdio.h>

int main()

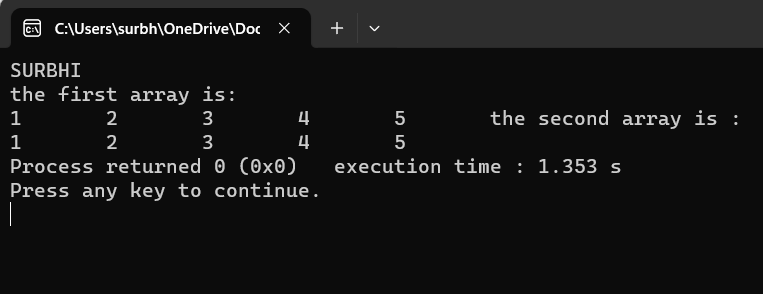
{

int n,i;

int a[5]={1,2,3,4,5};

int b[n];

printf("SURBHI\n");

 for(i=0;i<5;i++)

{

b[i] = a[i];

}

printf("the first array is:\n");

for(i=0;i<5;i++)

{

printf("%d\t",a[i]);

}

printf("the second array is :\n");

for (i=0;i<5;i++)

{

printf("%d\t",b[i]);

}

return 0;

}

**4. Write a program in C to separate odd and even integers into separate arrays:**

#include <stdio.h>

void main()

{

int arr1[10], arr2[10], arr3[10];

int i,j=0,k=0,n;

printf("\n\nSeparate odd and even integers in separate arrays:\n");

printf("------------------------------------------------------\n");

printf("Input the number of elements to be stored in the array :");

scanf("%d",&n);

printf("Input %d elements in the array :\n",n);

for(i=0;i<n;i++)

{

printf("element - %d : ",i);

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

}

for(i=0;i<n;i++)

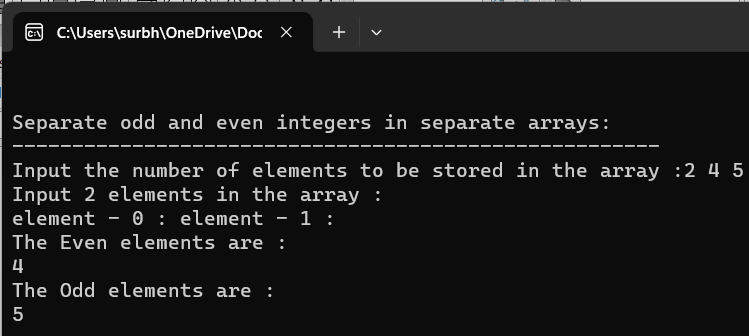
{

if (arr1[i]%2 == 0)

{

arr2[j] = arr1[i];

j++;

 }

else

{

arr3[k] = arr1[i];

k++;

}

}

printf("\nThe Even elements are : \n");

for(i=0;i<j;i++)

{

printf("%d ",arr2[i]);

}

printf("\nThe Odd elements are :\n");

for(i=0;i<k;i++)

{

printf("%d ", arr3[i]);

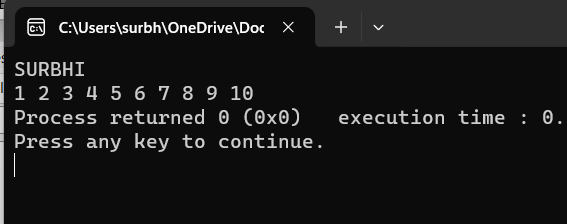
}

printf("\n\n");

}

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

#include<stdio.h>

int main()

{

int n,i;

int a[5]={1,2,3,4,5};

int b[5]={6,7,8,9,10};

int c[n];

printf("SURBHI\n");

printf("enter number of first array:\n");

for(i=0;i<5;i++)

{

scanf("%d",&a[5]);

}

printf("enter number of second array:\n");

for(i=0;i<5;i++)

{

scanf("%d",&b[5]);

}

printf("enter the elements after merging:\n");

for(i=0;i<5;i++)

{

c[n]=a[i];

c[n+5]=b[i];

}

for(i=0;i<10;i++)

{

printf("%d",c[n]);

}

return 0;

}

6. C Program to count the total number of duplicate array :

#include<stdio.h>

#define MAX\_SIZE 100

int main()

{

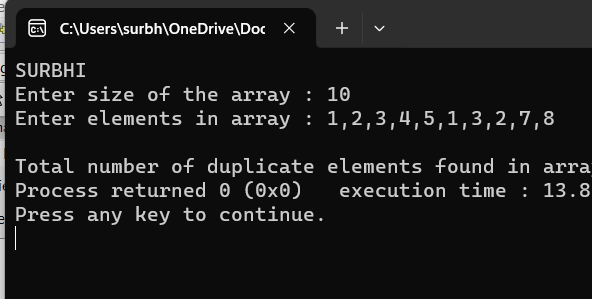
int arr[MAX\_SIZE];

int i,j,size,count=0;

printf("enter the size of array:\n");

scanf("%d",&size);

printf("enter the elements of first array:");

 for(i=0;i<size;i++)

{

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

}

for(i=0;i<size;i++)

{

for(j=i+1;j<size;j++)

{

if (arr[i] == arr[j]);

}

count++;

break;

}

printf("\n total number of duplicate array found in an array =%d",count);

return 0;

}

**7. Write a program in C to find the maximum and minimum elements in an array:**

#include <stdio.h>

#define MAX\_SIZE 100

int main()

{

int arr[MAX\_SIZE];

int i, max, min, size;

printf("SURNHI\n");

printf("Enter size of the array: ");

scanf("%d", &size);

printf("Enter elements in the array: ");

for(i=0; i<size; i++)

{

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

}

/\* Assume first element as maximum and minimum \*/

max = arr[0];

min = arr[0];

for(i=1; i<size; i++)

{

/\* If current element is greater than max \*/

if(arr[i] > max)

{

max = arr[i];

}

/\* If current element is smaller than min \*/

if(arr[i] < min)

{

min = arr[i];

}

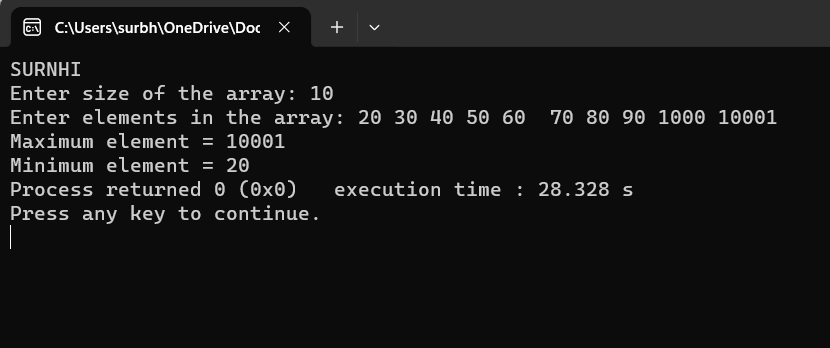
}

printf("Maximum element = %d\n", max);

printf("Minimum element = %d", min);

return 0;

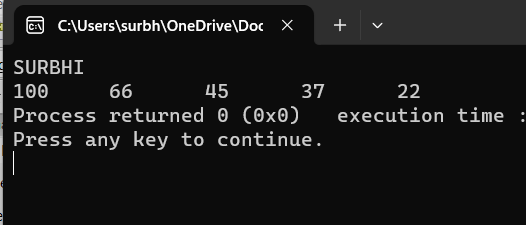
}



**8. Write a C program to sort the elements of an array in descending order:**

#include<stdio.h>

int main(){



int a[5] = {45,22,100,66,37};

int n = 5,i,j,t = 0;

printf("SURBHI\n");

for(i=0;i<n;i++)

{

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

{

if (a[i] < a[j])

{

t = a[i];

a[i] = a[j];

a[j] = t;

}

}

}

for(i=0;i<n;i++)

{

printf("%d\t",a[i]);

}

return 0;

}

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

#include<stdio.h>

int main( )

{

int a[50], b[25], i, j, k=1, s, m, n, temp ;

printf(" Enter the number of element in first array : ") ;

scanf("%d ",& m) ;

printf("\n Enter the element of first array in ascending order : \n") ;

for ( i = 1; i<=m; i++)

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

printf(" Enter the number of element in second array : ") ;

scanf("%d ",& n) ;

printf("\n Enter the element of second array in ascending order : \n") ;

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

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

s = m + n ;

for ( i = m+1 ; i <= s ; i++)

{

a[i] = b[k] ;

for ( j = 1 ; j <= s ; j++)

{

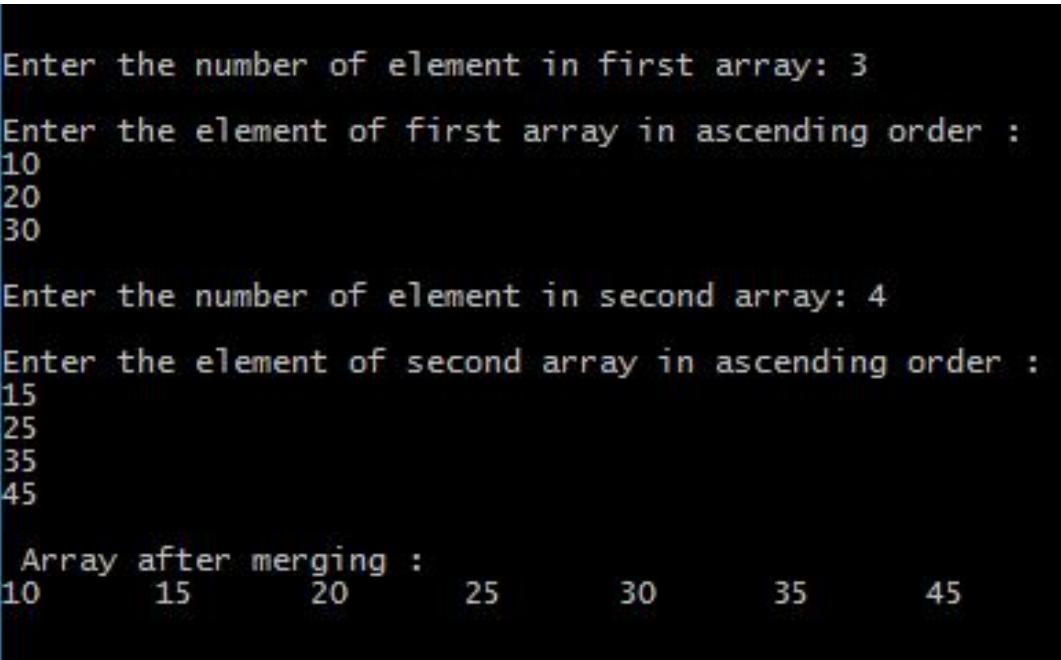
if ( a[j] >= a[i] )

{

temp = a[i] ;

a[j] = a[i] ;

a[i] = temp ;

}

}

k = k+1 ;

}

printf("\n Array after merging :\n") ;

for ( i = 1 ; i <= s ; i++)

printf("%d \t",a[i]) ;

return ( 0 ) ;

}

10.Consider two matrices of the size m and n. Implement matrix multiplication operation and display results using functions. Write three functions 1) Read matrix elements 2) Matrix Multiplication 3) Print matrix elements

#include <stdio.h>

#include <stdlib.h>

void readMatrix(int \*\*matrix, int rows, int cols) {

printf("SURBHI\n");

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

for (int j = 0; j < cols; ++j) {

printf("Enter element at position (%d, %d): ", i + 1, j + 1);

scanf("%d", &matrix[i][j]);

}

}

}

void printMatrix(int \*\*matrix, int rows, int cols) {

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

for (int j = 0; j < cols; ++j) {

printf("%d\t", matrix[i][j]);

}

printf("\n");

}

}

void multiplyMatrices(int \*\*A, int rowsA, int colsA, int \*\*B, int rowsB, int colsB, int \*\*result) {

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

for (int j = 0; j < colsB; ++j) {

result[i][j] = 0;

for (int k = 0; k < colsA; ++k) {

result[i][j] += A[i][k] \* B[k][j];

}

}

}

}

int main() {

int rowsM, colsM;

printf("Enter the number of rows and columns for Matrix M: ");

scanf("%d %d", &rowsM, &colsM);

int \*\*M = (int \*\*)malloc(rowsM \* sizeof(int \*));

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

M[i] = (int \*)malloc(colsM \* sizeof(int));

}

printf("Enter elements for Matrix M:\n");

readMatrix(M, rowsM, colsM);

int rowsN, colsN;

printf("Enter the number of rows and columns for Matrix N: ");

scanf("%d %d", &rowsN, &colsN);

int \*\*N = (int \*\*)malloc(rowsN \* sizeof(int \*));

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

N[i] = (int \*)malloc(colsN \* sizeof(int));

}

printf("Enter elements for Matrix N:\n");

readMatrix(N, rowsN, colsN);

if (colsM != rowsN) {

printf("Matrix multiplication is not possible.\n");

return 1;

}

int \*\*result = (int \*\*)malloc(rowsM \* sizeof(int \*));

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

result[i] = (int \*)malloc(colsN \* sizeof(int));

}

multiplyMatrices(M, rowsM, colsM, N, rowsN, colsN, result);

printf("Result of Matrix Multiplication:\n");

printMatrix(result, rowsM, colsN);

// Free allocated memory

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

free(M[i]);

}

free(M);

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

free(N[i]);

}

free(N);

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

free(result[i]);

}

free(result);

return 0;

}