

1. Take input of 3 integer numbers using scanf() function and calculate sum of them.

Solution :

```
#include<stdio.h>
int main()
{
    int x,y,z;
    printf("enter the first integer:\n");
    scanf("%d",&x);
    printf("enter the second integer:\n");
    scanf("%d",&y);
    printf("enter the third integer:\n");
    scanf("%d",&z);
    printf("\nsum is:%d",x+y+z);
    return 0;
}
```

2. Take a character input and print its corresponding ASCII value.

Solution:

```
#include<stdio.h>
int main()
{
    char ch;
    printf("enter any character\n");
    scanf("%c",&ch);
```

```
printf("\nASCII value of entered character is:%d",ch);  
    return 0;  
}
```

3.Take an integer ($0 < N < 128$) input and print its corresponding character.

Solution:

```
#include<stdio.h>  
int main()  
{  
    int x;  
    printf("enter any integer\n");  
    scanf("%d",&x);  
    printf("\ncorresponding character of entered  
integer is:%c",x);  
    return 0;  
}
```

4. Print a slash(/) and backslash(\) using printf() function.

Solution:

```
#include<stdio.h>  
int main()  
{  
    printf("%c",92);  
    printf("/");  
}
```

```
    return 0;
}
```

4. Take 11 inputs of any type of number and find the average of them.

Solution:

```
#include<stdio.h>
int main()
{
    int a[11],i,s;
    s=0;
    printf("enter 11 numbers\n");
    for(i=0;i<11;i++)
    {
        scanf("%d",&a[i]);
    }
    for(i=0;i<11;i++)
    {
        s=s+a[i];
    }
    printf("\nsum of eleven numbers are:%d",s);
    return 0;
}
```

5. Take two integer numbers. Calculate the sum, subtraction, product and division of them. Print the result in new lines.

Solution:

```
#include<stdio.h>
int main()
{
    int x,y,a,s,d,m;
    printf("enter the first number:\n");
    scanf("%d",&x);
    printf("enter the second number:\n");
    scanf("%d",&y);
    printf("\nsum is:%d",x+y);
    printf("\nsubtraction is:%d",x-y);
    printf("\nmultiplication is:%d",x*y);
    if(y!=0)
    {
        printf("\ndivision is:%f",(float)x/y);
    }
    else
    printf("\ninvalid input for division");
    return 0;
}
```

6. Take integer input using scanf() function and then take character input using scanf() and then print them.

Solution:

```
#include<stdio.h>
int main()
{
    char ch;
    int n;
    printf("enter a number:");
    scanf("%d",&n);
    printf("enter a character:");
    scanf("\n%c",&ch);
    printf("the number is:%d",n);
    printf("\nthe character is:%c",ch);
    return 0;
}
```

7. Suppose N students have appeared exam out of 300 marks. Consider marks obtained by them and determine percentage of marks obtained by individual student and the total students.

Solution:

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

```

float s[100],t;
int i,n;
t=0;
printf("enter the number of students:");
scanf("%d",&n);
for(i=0;i<n;i++)
{
    printf("marks of student number %d: ",i+1);
    scanf("%f",&s[i]);
}
for(i=0;i<n;i++)
{
    t=t+s[i];
}
for(i=0;i<n;i++)
{
    s[i]=s[i]*(.33);
}
for(i=0;i<n;i++)
{
    printf("\npercentage of marks of student %d is
%f",i+1,s[i]);
}
printf("\ntotal marks=%f",t);
return 0;
}

```

8. Consider two integers A and B. Now swap them.

Solution :

```
#include<stdio.h>
int main()
{
    int a,b,t;
    printf("enter first number:");
    scanf("%d",&a);
    printf("enter second number:");
    scanf("%d",&b);
    t=b;
    b=a;
    a=t;
    printf("first number after swap: %d",a);
    printf("\nsecond number after swap: %d",b);
    return 0;
}
```

9. Find maximum and minimum number between two integer numbers.

Solution:

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

```

int a,b;
printf("enter first number:");
scanf("%d",&a);
printf("enter second number:");
scanf("%d",&b);
if(a>b)
{
    printf("maximum number is %d",a);
    printf("\nminimum number is %d",b);
}
else
{
    printf("maximum number is %d",b);
    printf("\nminimum number is %d",a);
}
return 0;
}

```

10. Find maximum and minimum number between three integer numbers.

Solution:

```

#include<stdio.h>
int main()
{
    int a,b,c;

```



```
printf("enter 3 numbers: ");
scanf("%d%d%d",&a,&b,&c);
if(a>b&&a>c)
{
    printf("maximum number:%d\n",a);
}
else if(b>a&&b>c)
{
    printf("maximum number:%d\n",b);
}
else
{
    printf("maximum number:%d\n",c);
}
if(a<b&&a<c)
{
    printf("minimum number:%d\n",a);
}
else if(b<a&&b<c)
{
    printf("minimum number:%d\n",b);
}
else
{
    printf("minimum number:%d\n",c);
}
```

```
return 0;  
}
```

11. Find the maximum and minimum number among N given integers.

Solution:

```
#include<stdio.h>  
int main()  
{  
    int i,a,large,small,N;  
    printf("Enter total number of elements \n");  
    scanf("%d", &N);  
    printf("Enter first number \n");  
    scanf("%d", &a);  
    large=a;  
    small=a;  
    for(i=1;i<=N-1;i++)  
    {  
        printf("\nEnter next number:");  
        scanf("%d",&a);  
        if(a>large)  
            large=a;  
        if(a<small)  
            small=a;  
    }  
    printf("\nThe largest number is %d",large);
```

```
    printf("\nThe smallest number is %d",small);  
    return 0;  
}
```

12. Consider an integer and find out whether it is positive or negative or zero.

Solution:

```
#include<stdio.h>  
int main()  
{  
    int a;  
    printf("enter number:");  
    scanf("%d",&a);  
    if(a>0)  
        printf("%d is positive",a);  
    else if(a<0)  
        printf("%d is negative",a);  
    else  
        printf("ZERO");  
    return 0;  
}
```

13. Determine whether a number is odd or even.

Solution:

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

```
{
    int a;
    printf("enter number:");
    scanf("%d",&a);
    if(a%2==0)
        printf("%d is even",a);
    else
        printf("%d is odd",a);
    return 0;
}
```

14. Consider a character input and find out whether it is uppercase or lowercase.

Solution:

```
#include<stdio.h>
int main()
{
    char ch;
    printf("enter character:");
    scanf("%c",&ch);
    if(ch>='A'&&ch<='Z')
        printf("%c is uppercase",ch);
    else if(ch>='a'&&ch<='z')
        printf("%c is lowercase",ch);
    else
        printf("invalid input");
}
```

```
    return 0;
}
```

15. Take any input. Print the input if it is a character otherwise print that it is not character input.

Solution:

```
#include<stdio.h>
int main()
{
    char ch;
    printf("enter character:");
    scanf("%c",&ch);
    if((ch>='A'&&ch<='Z' )||(ch>='a'&&ch<='z'))
        printf("%c is character",ch);
    else
        printf("%c isnot character",ch);
    return 0;
}
```

16. Take two characters to check whether they are equal or not.

Solution:

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

```

char ch1,ch2;
printf("enter first character:");
scanf("%c",&ch1);
printf("enter second character:");
scanf("\n%c",&ch2);
if(ch1==ch2)
    printf("%c and %c characters are
equal",ch1,ch2);
else
    printf("%c and %c characters are not
equal",ch1,ch2);
return 0;
}

```

17. Enter two integers and check whether they are equal or not.

Solution:

```

#include<stdio.h>
int main()
{
    int a,b;
    printf("enter first integer:");
    scanf("%d",&a);
    printf("enter second integer:");
    scanf("%d",&b);
    if(a==b)

```

```
        printf("%d and %d integers are equal",a,b);
    else
        printf("%d and %d integers are not
equal",a,b);
    return 0;
}
```

18. Take two integers A and B. Check whether A is divisible by B or not.

Solution:

```
#include<stdio.h>
int main()
{
    int a,b;
    printf("enter first integer:");
    scanf("%d",&a);
    printf("enter second integer:");
    scanf("%d",&b);
    if(a%b==0)
        printf("%d is divisible by %d",a,b);
    else
        printf("%d is not divisible by %d",a,b);
    return 0;
}
```

19. Consider a year and determine whether it is leap year or not.

Solution:

```
#include <stdio.h>
int main()
{
    int year;
    printf("Enter a year to check if it is a leap
year\n");
    scanf("%d",&year);
    if(year%400==0)
        printf("%d is a leap year.\n",year);
    else if(year%100==0)
        printf("%d is not a leap year.\n",year);
    else if (year%4==0)
        printf("%d is a leap year.\n",year);
    else
        printf("%d is not a leap year.\n",year);
    return 0;
}
```

20. Take two integer and make the following menu:

- A. Addition
- B. Subtraction
- C. Multiplication

D. Division

Now take the user choice i.e. A, B etc and print the result.

Solution:

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

```
int main()
```

```
{
```

```
    int a,b;
```

```
    char ch;
```

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

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

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

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

```
    printf("enter A or a for addition\nenter S or s for  
subtraction\nenter M or m for  
multiplication\nenter D or d for division");
```

```
    printf("\nenter the type of operation:");
```

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

```
    switch(ch)
```

```
{
```

```
        case 'A':
```

```
        case 'a':
```

```
            printf("addition:%d",a+b);
```

```
            break;
```

```
        case 'S':
```

```
        case 's':
```

```

        printf("subtraction:%d",a-b);
        break;
    case 'M':
    case 'm':
        printf("multiplication:%d",a*b);
        break;
    case 'D':
    case 'd':
    {
        if(b!=0)
        {
            printf("%f",(float)a/b);
            break;
        }
        else
            printf("invalid input");
            break;
    }
    default:
        printf("invalid operation");
    }
    return 0;
}

```

21. Take two integer and make the following menu:

- (1) Addition
- (2) Subtraction
- (3) Multiplication
- (4) Division

Now take the user choice i.e. 1, 2 etc and print the result.

Solution:

```
#include<stdio.h>
int main()
{
    int a,b,i;
    printf("enter first number:");
    scanf("%d",&a);
    printf("enter second number:");
    scanf("%d",&b);
    printf("enter 1 for addition\nenter 2 for
    subtraction\nenter 3 for multiplication\nenter 4
    for division");
    printf("\nenter the type of operation:");
    scanf("\n%d",&i);
    switch(i)
    {
        case 1:
            printf("addition:%d",a+b);
            break;
        case 2:
```

```

        printf("subtraction:%d",a-b);
        break;
    case 3:
        printf("multiplication:%d",a*b);
        break;
    case 4:
    {
        if(b!=0)
        {
            printf("%f",(float)a/b);
            break;
        }
        else
            printf("invalid input");
            break;
    }
    default:
        printf("invalid operation");
    }
    return 0;
}

```

22. $1+2+3+\dots+N=?$ Take N as input integer.

Solution:

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

```
int main()
```

```

{
    int n,i,s;
    s=0;
    printf("enter the limit:");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        printf("%d",i);
        if(i!=n)
            printf("+");
        else
            printf("=");
        s=s+i;
    }
    printf("%d",s);
    return 0;
}

```

23. $1^2 + 2^2 + 3^2 + \dots + N^2 = ?$ Take N as input integer.

Solution:

```

#include<stdio.h>
int main()
{
    int n,i,s;
    s=0;
    printf("enter the limit:");

```

```

scanf("%d",&n);
for(i=1;i<=n;i++)
{
    printf("%d^2",i);
    if(i!=n)
        printf("+");
    else
        printf("=");
    s=s+(i*i);
}
printf("%d",s);
return 0;
}

```

24. $1+3+5+\dots+N=?$ Take N as input integer.

Solution:

```

#include<stdio.h>
int main()
{
    int n,i,s;
    s=0;
    printf("enter the limit:");
    scanf("%d",&n);
    for(i=1;i<=n;i=i+2)
    {
        printf("%d",i);
    }
}

```

```

        if(i==n||i==n-1)
            printf("=");
        else
            printf("+");
        s=s+i;
    }
    printf("%d",s);
    return 0;
}

```

25. $1 \cdot 2 + 2 \cdot 3 + 3 \cdot 5 + 4 \cdot 8 + 5 \cdot 12 + \dots = ?$ Take N as input integer.

Solution:

```

#include<stdio.h>
int main()
{
    int n,i,s,t2;
    printf("enter the limit:");
    scanf("%d",&n);
    s=0;t2=2;

    for(i=1;i<=n;i++)
    {
        printf("%d*%d",i,t2);
        if(i!=n)
            printf("+");
    }
}

```

```

        else
            printf("=");
            s=s+(i*t2);
            t2=t2+i;
        }
        printf("%d",s);
        return 0;
    }

```

26. $1+2-3+4-\dots+N=?$ Take N as input integer.

Solution:

```

#include<stdio.h>
int main()
{
    int s,i,a,b,n,f;
    printf("enter the limit:");
    scanf("%d",&n);
    s=1;
    f=1;
    for(i=2;i<=n;i++)
    {
        if(i%2==0)
        {
            a=i;

```



```

        b=0;
    }
    else
    {
        b=i;
        a=0;
    }
    b=(b*f);
    s+=a+b;
    f=f*(-1);
}
printf("sum:%d",s);
return 0;
}

```

27. Take an integer input N and find out the sum of the individual digit. Example: if N = 234 then your output will be $2+3+4 = 9$.

Solution:

```

#include<stdio.h>
int main()
{
    int n,r,s,t;
    printf("enter any number:");
    scanf("%d",&n);
    s=0;

```

```
t = n;
while (t != 0)
{
    r = t % 10;
    s = s + r;
    t = t / 10;
}
printf("sum of individual digits:%d",s);
return 0;
}
```

28. Find factorial value of input integer.

Solution:

```
#include<stdio.h>
int main()
{
    int n,i,f;
    printf("enter number:");
    scanf("%d",&n);
    f=1;
    for(i=1;i<=n;i++)
    {
        f=f*i;
    }
    printf("factorial value=%d",f);
    return 0;
}
```

```
}
```

29. Take any integer value. Find out whether it is prime or not.

Solution:

```
#include<stdio.h>
int main()
{
    int i,n,f;
    printf("enter any number:");
    scanf("%d",&n);
    f=0;
    for(i=2;i<=n/2;i++)
    {
        if(n%i==0)
        {
            f=1;
            break;
        }
    }
    if(f==0)
        printf("%d is prime number",n);
    else
        printf("%d isnot prime number",n);
    return 0;
}
```

30. Take an integer input and print its all factors.

Solution:

```
#include<stdio.h>
int main()
{
    int i,n;
    printf("enter number:");
    scanf("%d",&n);
    printf("the factors are:\n");
    for(i=1;i<=n;i++)
    {
        if(n%i==0)
            printf("%d\n",i);
    }
    return 0;
}
```

31. Print the fibonacci series.

Solution:

```
#include<stdio.h>
int main()
{
    int f1,f2,f,i,n;
    printf("enter the limit:");
    scanf("%d",&n);
```

```

f1=0;
f2=1;
f=f1+f2;
printf("the series:");
printf("\n%d\n%d",f1,f2);
while(f<n)
{
    f=f1+f2;
    f1=f2;
    f2=f;
    printf("\n%d",f);
}
return 0;
}

```

32. Print the sinx series.

Solution:

```

#include<stdio.h>
#include<math.h>
#define PI acos(-1)
int main()
{
    double x, nr, dr, fact, term, temp_term,
term_sign, sum;
    printf("enter the value:");
    scanf("%lf", &x);

```

```

x=(x*PI)/180;
dr=1;
nr=x;
sum=x;
term_sign=-1;
for(term=1 ; term<2*7 ; term+=2)
{
    nr=nr*x*x;
    dr=dr*(term+1)*(term+2);
    temp_term=nr/dr;
    temp_term=temp_term*term_sign;
    sum=sum+temp_term;
    term_sign=term_sign*-1;
}
printf("value:%lf", sum);
return 0;
}

```

33. Print the cosx series.

Solution:

```

#include<stdio.h>
#include<math.h>
#define PI acos(-1)
int main()
{

```

```

        double x, nr, dr, fact, term, temp_term,
term_sign, sum;
    printf("enter the value:");
    scanf("%lf", &x);
    x=(x*PI)/180;
    dr=1;
    nr=1;
    sum=1;
    term_sign=-1;
    for(term=0 ; term<2*20 ; term+=2)
    {
        nr=nr*x*x;
        dr=dr*(term+1)*(term+2);
        temp_term=nr/dr;
        temp_term=temp_term*term_sign;
        sum=sum+temp_term;
        term_sign=term_sign*-1;
    }
    printf("value: %lf", sum);
    return 0;
}

```

34. Take an integer input and check whether it is perfect number or not.

Solution:

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

```

int main()
{
    int s=0,n,i=1;
    scanf("%d",&n);
    while(i<n)
    {
        if(n%i==0)
            s=s+i;
        i++;
    }
    if(s==n)
        printf("perfect");
    else
        printf("not perfect");
    return 0;
}

```

35. Take a binary input and convert to decimal
Solution:

```

#include<stdio.h>
int main()
{
    int b,d=0,p=1,r;
    printf("enter a binary number:");
    scanf("%d",&b);
    while(b!=0){

```



```

        r=b%10;
        d=d+r*p;
        p=p*2;
        b=b/10;
    }
    printf("the decimal value is: %d",d);
return 0;
}

```

36. Take a decimal input and convert it to binary.

Solution:

```

#include<stdio.h>
int main()
{
    long int d,r,q;
    int b[100],i,j;
    i=1;
    printf("enter the decimal number:");
    scanf("%ld",&d);
    q=d;
    while(q!=0)
    {
        b[i++]=q%2;
        q=q/2;
    }
    printf("the binary value is:");

```

```

    for(j=i-1;j>0;j--)
    {
        printf("%d",b[j]);
    }
    return 0;
}

```

37. Take a decimal input and convert to octal.

Solution:

```

#include<stdio.h>
int main()
{
    Long int d,r,q;
    int o[100],i,j;
    i=1;
    printf("enter the deciaml number:");
    scanf("%ld",&d);
    q=d;
    while(q!=0)
    {
        o[i++]=q%8;
        q=q/8;
    }
    printf("the octal value is:");
    for(j=i-1;j>0;j--)
    {

```

```

        printf("%d",o[j]);
    }
    return 0;
}

```

38. Take decimal input and convert to hexadecimal

Solution:

```

#include<stdio.h>
int main()
{
    long int d,r,q;
    int h[100],i,j,t;
    i=1;
    printf("enter the decimal number:");
    scanf("%ld",&d);
    q=d;
    while(q!=0)
    {
        t=q%16;
        if(t<10)
            t=t+48;
        else
            t=t+55;
        h[i++]=t;
        q=q/16;
    }
}

```

```

    }
    printf("the hexadecimal value is:");
    for(j=i-1;j>0;j--)
    {
        printf("%c",h[j]);
    }
    return 0;
}

```

39. Take octal input and convert to decimal.

Solution:

```

#include<stdio.h>
#include<math.h>
int main()
{
    long int o,d;
    int i;
    i=0;
    d=0;
    printf("enter the octal number:");
    scanf("%ld",&o);
    while(o!=0)
    {
        d=d+(o%10)*pow(8,i++);
        o=o/10;
    }
}

```

```
    printf("the decimal value is:");  
    printf("%ld",d);  
    return 0;  
}
```

40. Take octal input and convert to binary.

Solution:

```
#include<stdio.h>  
int main()  
{  
    char o[100];  
    long int i;  
    i=0;  
    printf("enter octal number:");  
    scanf("%s",&o);  
    printf("the binary value is:");  
    while(o[i])  
    {  
        switch(o[i])  
        {  
            case '0':  
                printf("000");  
                break;  
            case '1':  
                printf("001");  
                break;
```

```
        case '2':
            printf("010");
            break;
        case '3':
            printf("011");
            break;
        case '4':
            printf("100");
            break;
        case '5':
            printf("101");
            break;
        case '6':
            printf("110");
            break;
        case '7':
            printf("111");
            break;
        default:
            printf("\ninvalid octal digit %c",o[i]);
            return 0;
    }
    i++;
}
return 0;
}
```

41. Take hexadecimal input and convert to binary.

Solution:

```
#include<stdio.h>
int main()
{
    char b[100],h[100];
    long int i;
    i=0;
    printf("enter hexadecimal number:");
    scanf("%s",&h);
    printf("the binary value is:");
    while(h[i])
    {
        switch(h[i])
        {
            case '0':
                printf("0000");
                break;
            case '1':
                printf("0001");
                break;
            case '2':
                printf("0010");
                break;
            case '3':
```

```
        printf("0011");
        break;
case '4':
    printf("0100");
    break;
case '5':
    printf("0101");
    break;
case '6':
    printf("0110");
    break;
case '7':
    printf("0111");
    break;
case '8':
    printf("1000");
    break;
case '9':
    printf("1001");
    break;
case 'A':
    printf("1010");
    break;
case 'B':
    printf("1011");
    break;
```



```
case 'C':  
    printf("1100");  
    break;  
case 'D':  
    printf("1101");  
    break;  
case 'E':  
    printf("1110");  
    break;  
case 'F':  
    printf("1111");  
    break;  
case 'a':  
    printf("1010");  
    break;  
case 'b':  
    printf("1011");  
    break;  
case 'c':  
    printf("1100");  
    break;  
case 'd':  
    printf("1101");  
    break;  
case 'e':  
    printf("1110");
```

```

        break;
    case 'f':
        printf("1111");
        break;
    default:
        printf("\ninvalid hexadecimal digit
%c",h[i]);
        return 0;
    }
    i++;
}
return 0;
}

```

42. Take binary input and convert to octal

Solution:

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

```
int main()
```

```
{
```

```
    long int binarynum, octalnum = 0, j = 1,
    remainder;
```

```
    printf("Enter the value for binary number: ");
```

```
    scanf("%ld", &binarynum);
```

```
    while (binarynum != 0)
```

```
    {
```

```
        remainder = binarynum % 10;
```

```

        octalnum = octalnum + remainder * j;
        j = j * 2;
        binarynum = binarynum / 10;
    }
    printf("Equivalent octal value: %lo", octalnum);
    return 0;
}

```

43. Take binary input and convert to hexadecimal.

Solution:

```

#include <stdio.h>
int main()
{
    long int binaryval, hexadecimalval = 0, i = 1,
    remainder;
    printf("Enter the binary number: ");
    scanf("%ld", &binaryval);
    while (binaryval != 0)
    {
        remainder = binaryval % 10;
        hexadecimalval = hexadecimalval + remainder
        * i;
        i = i * 2;
        binaryval = binaryval / 10;
    }
}

```

```

        printf("Equivalent hexadecimal value: %lX",
hexadecimalval);
    return 0;
}

```

44. Take two binary number as input and find sum of them.

Solution:

```

#include <stdio.h>
int main()
{
    long binary1, binary2;
    int i = 0, remainder = 0, sum[20];
    printf("Enter the first binary number: ");
    scanf("%ld", &binary1);
    printf("Enter the second binary number: ");
    scanf("%ld", &binary2);
    while (binary1 != 0 || binary2 != 0)
    {
        sum[i++] =(binary1 % 10 + binary2 % 10 +
remainder) % 2;
        remainder =(binary1 % 10 + binary2 % 10 +
remainder) / 2;
        binary1 = binary1 / 10;
        binary2 = binary2 / 10;
    }
}

```

```

    if (remainder != 0)
        sum[i++] = remainder;
    --i;
    printf("Sum of two binary numbers: ");
    while (i >= 0)
        printf("%d", sum[i--]);
    return 0;
}

```

45. Declare an array of integers and find the maximum number.

Solution:

```

#include<stdio.h>
int main()
{
    int ar1[100];
    int i,max,n;
    printf("enter number of elements of array:");
    scanf("%d",&n);
    printf("enter the numbers:");
    for(i=0;i<n;i++)
    {
        scanf("%d",&ar1[i]);
    }
    max=ar1[0];
    for(i=0;i<n;i++)

```

```

{
    if(max<ar1[i])
        max=ar1[i];
}
printf("the maximum number is:%d",max);
return 0;
}

```

46. Declare an array of integers and find the minimum number.

Solution:

```

#include<stdio.h>
int main()
{
    int ar1[100];
    int i,min,n;
    printf("enter number of elements of array:");
    scanf("%d",&n);
    printf("enter the numbers:");
    for(i=0;i<n;i++)
    {
        scanf("%d",&ar1[i]);
    }
    min=ar1[0];
    for(i=0;i<n;i++)
    {

```

```

        if(min>ar1[i])
            min=ar1[i];
    }
    printf("the minimum number is:%d",min);
    return 0;
}

```

47. Declare the array of integers. Remove the common elements from the first array and print the rest of the numbers.

Solution:

```

#include <stdio.h>
int main()
{
    int inputArray[500], elementCount, counter;
    int readIndex, writeIndex, scanIndex;
    printf("Enter number of elements in array: ");
    scanf("%d", &elementCount);
    printf("Enter %d numbers \n", elementCount);
    for(counter = 0; counter < elementCount;
counter++)
    {
        scanf("%d", &inputArray[counter]);
    }
    for(readIndex=0, writeIndex=0; readIndex <
elementCount; readIndex++)

```

```

    {
        for(scanIndex=readIndex+1; scanIndex <
elementCount; scanIndex++)
        {
            if(inputArray[scanIndex] ==
inputArray[readIndex])
            {
                break;
            }
        }
        if(scanIndex == elementCount)
        {
            inputArray[writeIndex] =
inputArray[readIndex];
            writeIndex++;
        }
    }
    printf("Unique Elements\n");
    for(counter = 0; counter < writeIndex;
counter++)
    {
        printf("%d ", inputArray[counter]);
    }
    return 0;
}

```


48. Determine the second maximum element in an array of integers.

Solution:

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

```
int main()
```

```
{
```

```
    int inputArray[500], elementCount, counter;
```

```
    int maxElement, secondMaxElement, INT_MIN;
```

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

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

```
    if(elementCount < 2)
```

```
    {
```

```
        printf("Number of elements should be more  
than or equal to two");
```

```
        return 0;
```

```
    }
```

```
    printf("Enter %d numbers \n", elementCount);
```

```
    for(counter = 0; counter < elementCount;  
counter++)
```

```
    {
```

```
        scanf("%d", &inputArray[counter]);
```

```
    }
```

```
    for(counter = 0; counter < elementCount;  
counter++)
```

```
    {
```

```
        if(inputArray[counter] > maxElement)
```

```

        {
            secondMaxElement = maxElement;
            maxElement = inputArray[counter];
        }

        else if (inputArray[counter] >
secondMaxElement && inputArray[counter] !=
maxElement)
        {
            secondMaxElement =
inputArray[counter];
        }
    }

    printf("Second Maximum element: %d",
secondMaxElement);
    return 0;
}

```

49. Take input integers in an array and print them in reverse order.

Solution:

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

```
int main()
```

```
{
```

```
    int inputArray[500], outputArray[500],
    elementCount, counter;
```

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

```

scanf("%d", &elementCount);
printf("Enter %d numbers \n", elementCount);
    for(counter = 0; counter < elementCount;
counter++)
    {
        scanf("%d", &inputArray[counter]);
    }
    for(counter = 0; counter < elementCount;
counter++)
    {
        outputArray[counter] =
inputArray[elementCount-counter-1];
    }
    printf("Reversed Array\n");
    for(counter = 0; counter < elementCount;
counter++){
        printf("%d ", outputArray[counter]);
    }
    return 0;
}

```

50. Insert element in an array of integers.

Solution:

```

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

```

```

    int inputArray[500], elementCount, counter,
    element, index;
    printf("Enter number of elements in array: ");
    scanf("%d", &elementCount);
    printf("Enter %d numbers \n", elementCount);
    for(counter = 0; counter < elementCount;
counter++)
    {
        scanf("%d", &inputArray[counter]);
    }
    printf("Enter number to be inserted\n");
    scanf("%d", &element);
    printf("Enter position where you want to insert
an element\n");
    scanf("%d", &index);
    for(counter = elementCount; counter > index-1;
counter--)
    {
        inputArray[counter] = inputArray[counter-1];
    }
    inputArray[index-1] = element;
    printf("Updated Array\n");
    for(counter = 0; counter < elementCount + 1;
counter++)
    {
        printf("%d ", inputArray[counter]);

```

```
    }  
    return 0;  
}
```

51. Declare an array where you can put your name.

Solution:

```
#include<stdio.h>  
#include<string.h>  
int main()  
{  
    char n[100],ch;  
    int i;  
    i=0;  
    printf("INPUT: ");  
    do  
    {  
        ch=getchar();  
        n[i]=ch;  
        i++;  
    }  
    while(ch!='\n');  
    n[i]='\0';  
    printf("OUTPUT: ");  
    printf("%s",n);  
    return 0;
```

```
}
```

52. Take a string and determine whether it is palindrome or not.

Solution:

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

```
int main()
```

```
{
```

```
    char a[100];
```

```
    int count,i,len,flag=0;
```

```
    count=0;
```

```
    printf("\nENTER A STRING: ");
```

```
    gets(a);
```

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

```
    {
```

```
        count++;
```

```
    }
```

```
    for (i=0;i<count;i++)
```

```
    {
```

```
        if(a[i]==a[count-i-1])
```

```
            flag=flag+1;
```

```
    }
```

```
    if(flag==count)
```

```
        printf("\nTHE STRING IS PALINDROME");
```

```

else
    printf("\nTHE STRING IS NOT
PALINDROME");
    return 0;
}

```

53. Take an integer and determine whether it is palindrome or not.

Solution:

```

#include <stdio.h>
int main()
{
    int n, reversedInteger = 0, remainder,
originalInteger;
    printf("Enter an integer: ");
    scanf("%d", &n);
    originalInteger = n;
    while( n!=0 )
    {
        remainder = n%10;
        reversedInteger = reversedInteger*10 +
remainder;
        n /= 10;
    }
    if (originalInteger == reversedInteger)
        printf("%d is a palindrome.", originalInteger);
}

```

```

        else
            printf("%d is not a palindrome.",
originalInteger);
        return 0;
    }

```

54. Take two strings and print 0 if both are equal, 1 if first string is greater than second and -1 if second string is greater than first.

Solution:

```

#include<stdio.h>
#include<math.h>
int main()
{
    int a,b,co,i,j,l,m;
    a=0;b=0;
    co=0;
    char s[100];
    char c[100];
    printf("enter first string:");
    gets(s);
    printf("enter second string:");
    gets(c);
    l= strlen(s);
    m=strlen(c);
    int k;

```



```

if(m < l)
    k = m;
else
    k = l;
for( i = 0; i < k; i++)
{
    if(s[i] < c[i])
    {
        printf("-1");
        return 0;
    }
    else if(c[i] < s[i])
    {
        printf("1");
        return 0;
    }
}
if(l < m) printf("-1");
else if(l > m) printf("1");
else printf("0");
return 0;
}

```

55. Take 3×3 matrix (2 dimensional array) as input and then print it.

Solution:

```
#include <stdio.h>
int main()
{
    int array[10][10];
    int i, j, m, n;
    printf("Enter the order of the matrix \n");
    scanf("%d %d", &m, &n);
    printf("Enter the coefficients of the matrix\n");
    for (i = 0; i < m; ++i)
    {
        for (j = 0; j < n; ++j)
        {
            scanf("%d", &array[i][j]);
        }
    }
    printf("The given matrix is \n");
    for (i = 0; i < m; ++i)
    {
        for (j = 0; j < n; ++j)
        {
            printf(" %d", array[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

56. Find the GCD and LCD of two given inputs.

Solution:

```
#include<stdio.h>
#include<math.h>
int main()
{
    int a,b,x,y,temp;
    printf("enter first number:");
    scanf("%d",&x);
    printf("enter second number:");
    scanf("%d",&y);
    a=x;
    b=y;
    while(b!=0)
    {
        temp=b;
        b=a%b;
        a=temp;
    }
    printf("GCD:%d \nLCM:%d",a,(x*y)/a);
    return 0;
}
```

57. Find the GCD and LCM among 3 integers.

Solution:

```
#include<stdio.h>
int main()
{
    int a,b,c,l,g,p,i;
    printf("Enter three numbers: ");
    scanf("%d%d%d",&a,&b,&c);
    i=1;
    while(i<=a)
    {
        if(a%i==0&&b%i==0&&c%i==0)
        {
            g=i;
            i++;
        }
        i=1;
        p=a*b*c;
        while(c<=p)
        {
            if(i%a==0&&i%b==0&&i%c==0)
            {
                l=i;
                break;
            }
            i++;
        }
        printf("GCD: %d\nLCM: %d",g,l);
        return 0;
    }
```

```
}
```

58. Convert into celsius scale from fahrenheit scale.

Solution:

```
#include<stdio.h>
int main()
{
    float fh,cl;
    printf("Enter temperature value in Fahrenheit:
");
    scanf("%f", &fh);
    cl = (fh - 32) / 1.8;
    printf("Converted Celsius value: %f",cl);
    return 0;
}
```

59. Convert into fahrenheit scale from celsius scale.

Solution:

```
#include <stdio.h>
int main()
{
    float celsius, fahrenheit;
    printf("Enter temperature in Celsius: ");
    scanf("%f", &celsius);
```

```

    fahrenheit = ((celsius * 9)/5) + 32;
    printf("\n%.2f Celsius = %.2f Fahrenheit",
celsius, fahrenheit);
    return 0;
}

```

60. $2+4+6+\dots+N=?$ Take N as input.

Solution:

```

#include<stdio.h>
int main()
{
    int n,i,s;
    s=0;
    printf("enter the limit:");
    scanf("%d",&n);
    for(i=2;i<=n;i++)
    {
        if(i%2==0)
        {
            printf("%d",i);
            if(i==n||i==n-1)
                printf("=");
            else
                printf("+");
            s=s+i;
        }
    }
}

```

```
}  
printf("%d",s);  
return 0;  
}
```

61. Print and add the even and odd numbers from N numbers where N is given input.

Solution:

```
#include<stdio.h>  
int main()  
{  
    int e[100],o[100],i,j,n,k,m;  
    k=0;  
    m=0;  
    printf("enter the range of number:");  
    scanf("%d",&n);  
    for(i=1;i<=n;i++)  
    {  
        if(i%2==0)  
        {  
            e[k]=i;  
            k++;  
        }  
        else  
        {  
            o[m]=i;
```

```

        m++;
    }
}
printf("even numbers:");
for(i=0;i<k;i++)
{
    printf("%d ",e[i]);
}
printf("\nodd numbers:");
for(i=0;i<m;i++)
{
    printf("%d ",o[i]);
}
return 0;
}

```

62. Determine the roots of binomial equation
 $ax^2 + bx + c = 0$.

Solution:

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

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

```
int main()
```

```
{
```

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

```
        double
```

```
    realPart,imaginaryPart,determinant,root1,root2;
```



```

printf("enter roots of the equation:\n");
scanf("%d %d %d",&a,&b,&c);
determinant=(b*b)-(4*a*c);
if (determinant==0)
{
    root1=root2=-(b/(2*a));
    printf("the required roots are:\n%lf\n %lf",
root1,root2);
}
else if(determinant>0)
{
    root1=(-b+sqrt(determinant))/(2*a);
    root2=(-b-sqrt(determinant))/(2*a);
    printf("the required roots
are:\n%lf\n%lf",root1,root2);
}
else
{
    realPart=(-b/(2*a));
    imaginaryPart=sqrt(-determinant)/(2*a);
    printf("the required roots are:\n%lf+%lfi\n
%lf-%lfi",realPart,imaginaryPart,realPart,imaginary
Part);
}
return 0;
}

```

63. Determine the area of a triangle whose length of three sides are given.

Solution:

```
#include <stdio.h>
#include <math.h>
void main()
{
    float s, a, b, c, area;
    printf("Enter the values of a, b and c \n");
    scanf("%f %f %f", &a, &b, &c);
    s = (a + b + c) / 2;
    area = sqrt(s * (s - a) * (s - b) * (s - c));
    printf("Area of a triangle = %f", area);
    return 0;
}
```

64. Determine the area of a triangle whose length of base and height are given.

Solution:

```
#include <stdio.h>
#include <math.h>
void main()
{
    float h,b,area;
    printf("Enter the values base and height:");
```

```
scanf("%f %f",&b,&h);  
area = (b*h) / 2;  
printf("Area of a triangle = %f", area);  
return 0;  
}
```

65. Determine the area of parallelogram.

Solution:

```
#include <stdio.h>  
#include <math.h>  
void main()  
{  
    float h,b,area;  
    printf("Enter the values base and height:");  
    scanf("%f %f",&b,&h);  
    area = (b*h);  
    printf("Area of a parallelogram = %f", area);  
    return 0;  
}
```

66. Determine the area of rhombus.

Solution:

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

```
float h,b,area;
printf("Enter the values of two diagonals:");
scanf("%f %f",&b,&h);
area = (b*h)/2;
printf("Area of a rhombus = %f", area);
return 0;
}
```

67. Determine the area of rectangle.

Solution:

```
#include <stdio.h>
#include <math.h>
void main()
{
    float h,b,area;
    printf("Enter the values base and height:");
    scanf("%f %f",&b,&h);
    area = (b*h);
    printf("Area of a rectangle = %f", area);
    return 0;
}
```

68. Determine the perimeter of rectangle.

Solution:

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

```

void main()
{
    float h,b,p;
    printf("Enter the values of two sides:");
    scanf("%f %f",&b,&h);
    p = (b+h)*2;
    printf("perimeter of a rectangle = %f", p);
    return 0;
}

```

69. Determine the number of positive and negative numbers from N number of inputs.

Solution:

```

#include<stdio.h>
int main()
{
    int a[100],p[100],ne[100],i,j,n,k,m,f1,f2;
    k=0;
    m=0;
    f1=0;
    f2=0;
    printf("enter the number of numbers:");
    scanf("%d",&n);
    printf("enter the values:");
    for(i=0;i<n;i++)
    {

```

```
        scanf("%d",&a[i]);
    }
    for(i=0;i<n;i++)
    {
        if(a[i]>=0)
        {
            p[k]=a[i];
            k++;
            f1++;
        }
        else
        {
            ne[m]=a[i];
            m++;
            f2++;
        }
    }
    printf("number of positive numbers is %d",f1);
    printf("\npositive numbers are:");
    for(i=0;i<k;i++)
    {
        printf("%d ",p[i]);
    }

    printf("\nnumber of negative numbers is
%d",f2);
    printf("\nnegative numbers:");
```

```

    for(i=0;i<m;i++)
    {
        printf("%d ",ne[i]);
    }
    return 0;
}

```

70. $1 + \frac{1}{3} + \frac{1}{5} + \dots + \frac{1}{N} = ?$ Take N as input integer.

Solution:

```

#include<stdio.h>
int main()
{
    double n,sum=0,i;
    printf("Please Give The Value of N: ");
    scanf("%lf",&n);
    for(i=1;i<=n;i++)
    {
        sum = sum + (1/i);
    }
    printf("sum=%.2lf",sum);
    return 0;
}

```

71. Apply the binary search method to find out any definite number.

Solution:

```
#include <stdio.h>
int main()
{
    int c, first, last, middle, n, search, array[100];
    printf("Enter number of elements\n");
    scanf("%d",&n);
    printf("Enter %d integers\n", n);
    for (c = 0; c < n; c++)
    {
        scanf("%d",&array[c]);
    }
    printf("Enter value to find\n");
    scanf("%d", &search);
    first = 0;
    last = n - 1;
    middle = (first+last)/2;
    while (first <= last)
    {
        if (array[middle] < search)
            first = middle + 1;
        else if (array[middle] == search)
        {
            printf("%d found at location %d.\n", search,
middle+1);
            break;
        }
    }
}
```



```

        else
            last = middle - 1;
            middle = (first + last)/2;
    }
    if (first > last)
        printf("Not found! %d is not present in the
list.\n", search);
    return 0;
}

```

72. $(1 \times 2) + (2 \times 3) + \dots + N \times (N + 1) = ?$ Take N as input

Solution:

```

#include<stdio.h>
int main()
{
    int i,n,s;
    s=0;
    printf("enter the range:");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        printf("(%d*%d)",i,i+1);
        if(i==n)
            printf("=");
        else

```

```
        printf("+");
        s=s+(i*(i+1));
    }
    printf("%d",s);
    return 0;
}
```

73. Prepare a program to determine the grade sheet of the students of a class.

Solution:

```
#include<stdio.h>
int main()
{
    float n,a[100];
    int i;
    printf("enter the number of students:");
    scanf("%f",&n);
    printf("enter the marks of students:");
    for(i=0;i<n;i++)
    {
        scanf("%f",&a[i]);
    }
    for(i=0;i<n;i++)
    {
        if(a[i]>=90)
        {
```

```
    printf("\nstudent %d got A+",i+1);
}
else if(a[i]<90&&a[i]>=80)
{
    printf("\nstudent %d got A",i+1);
}
else if(a[i]<80&&a[i]>=70)
{
    printf("\nstudent %d got A-",i+1);
}
else if(a[i]<70&&a[i]>=60)
{
    printf("\nstudent %d got B",i+1);
}
else if(a[i]<60&&a[i]>=50)
{
    printf("\nstudent %d got C",i+1);
}
else if(a[i]<50&&a[i]>=40)
{
    printf("\nstudent %d got D",i+1);
}
else
    printf("\nstudent %d got F",i+1);
}
return 0;
```

```
}
```

74. Print the multiplication table of N number.

Solution:

```
#include<stdio.h>
int main()
{
    int i,x;
    printf("enter :");
    scanf("%d",&x);
    for(i=1;i<=10;i++)
    {
        printf("%d*%d=%d\n",x,i,x*i);
    }
    return 0;
}
```

75. Take input of 1-N numbers and print the numbers excluding anyone among them.

Solution:

```
#include<stdio.h>
int main()
{
    int n,i,x;
    printf("enter number range:");
    scanf("%d",&n);
```

```

printf("number to be excluded:");
scanf("%d",&x);
printf("numbers after exclusion:");
for(i=1;i<=n;i++)
{
    if(i!=x)
        printf("%d ",i);
}
return 0;
}

```

76. Determine the value of X^N

Solution:

```

#include<stdio.h>
int main()
{
    int p,n,i,x;
    p=1;
    printf("enter value of X:");
    scanf("%d",&x);
    printf("enter value of N:");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        p=p*x;
    }
}

```

```
    printf("value of %d^%d=%d",x,n,p);  
    return 0;  
}
```

77. Determine the unit and tenth place digit of a given number.

Solution:

```
#include<stdio.h>  
int main()  
{  
    int n;  
    printf("enter the number:");  
    scanf("%d",&n);  
    printf("unit place:%d",n%10);  
    printf("\nten's place:%d",(n/10)%10);  
    return 0;  
}
```

78. Given the three angles of a triangle. Find whether the triangle is valid or not.

Solution:

```
#include<stdio.h>  
int main()  
{  
    float x,y,z;  
    printf("enter the three angles:");
```

```
scanf("%f%f%f",&x,&y,&z);  
if(x+y+z<=180)  
    printf("triangle is valid");  
else  
    printf("triangle is not valid");  
return 0;  
}
```

79.

