1. Printing patterns :

\*\*\*\*\*\*\*  
\*\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*  
\*\*\*  
\*\*  
\*

#include <stdio.h>  
int main() {  
 int n;  
 printf("Please enter number of starts line value \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ;i<=n ; i++){  
   
   
 for( int k = i ;k<=n ; k++){  
   
 printf("\*");  
 }  
 printf("\n");  
 }  
   
}

1. Printing patterns :

\*  
\*\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*

#include <stdio.h>  
int main() {  
 int n;  
 printf("Please enter number of starts line value \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ;i<=n ; i++){  
   
   
 for( int k = 1 ;k<=i ; k++){  
   
 printf("\*");  
 }  
 printf("\n");  
 }  
   
}

1. Printing patterns :

-------\*  
------\*\*  
-----\*\*\*  
----\*\*\*\*  
---\*\*\*\*\*  
--\*\*\*\*\*\*  
-\*\*\*\*\*\*\*

#include <stdio.h>  
int main() {  
 int n;  
 printf("Please enter number of starts line value \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ;i<=n ; i++){  
   
 for( int j = i ;j<=n ; j++){  
   
 printf("-");  
 }  
   
 for( int k = 1 ;k<=i ; k++){  
   
 printf("\*");  
 }  
 printf("\n");  
 }  
   
}

1. Printing patterns :

\*  
MM§M \*\*\*  
MM§M \*\*\*\*\*  
MM§M \*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
MM§M \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <stdio.h>  
int main() {  
int n,i,j,k,c=80;  
printf("Please enter number of starts line value \n");  
scanf("%d",&n);  
  
for( int i = 1 ;i<=n ; i++){  
for( int j = 1 ;j<=c/2-i ; j++){  
printf(" ");  
}  
for( int k = 1 ;k<=2\*i-1 ; k++){  
printf("\*");  
}  
printf("\n");  
}  
}

1. Printing patterns :

\*\*\*\*\*-  
\*\*\*\*--  
\*\*\*---  
\*\*----  
\*-----

#include <stdio.h>  
int main() {  
 int n;  
 printf("Please enter number of starts line value \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ;i<=n ; i++){  
   
 for( int j = i ;j<=n ; j++){  
   
 printf("\*");  
 }  
   
 for( int k = 1 ;k<=i ; k++){  
   
 printf("-");  
 }  
 printf("\n");  
 }  
   
}

1. Find the factorial for a given number:

#include <stdio.h>  
int main() {  
int n,i,j,k,su=80;  
printf("Please enter n value \n");  
scanf("%d",&n);  
int fact =1 ;  
for( int i = 1 ;i<=n ; i++){  
  
fact =fact \* i ;  
  
}  
printf("%d is the factorial for the number you entered”, fact);  
  
}

1. add the giving digit number code :

#include <stdio.h>  
int main() {  
int n,sum=0;  
printf("Please enter n value \n");  
scanf("%d",&n);  
int r =n%10 ;  
   
while (n>0){  
int r =n%10;  
sum = sum+r ;  
n= n/10;  
}  
  
printf("%d the sum for the number you entered”, sum);  
  
}

1. find if number is a Perfect number:

any number factor sum except itself is equal to the number

Diagram

Description automatically generated#include <stdio.h>  
int main() {  
 int n ,i ,sum=0;  
 printf("Please enter number that you want to check if its perfect number \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ; i<n ; i++){  
   
 if (n%i==0){  
 sum =sum+i ;  
   
   
 }  
   
 }  
 if(n==sum){  
 printf("%d , is a perfect number",n);  
   
 }  
   
 else{  
 printf("%d , is not a perfect number.",n);  
   
 }   
   
}

1. Fibonacci series: 0,1,1,2,3,5,8,13

#include <stdio.h>  
int main() {  
 int n ,a=0,b=1,c;  
 printf("Please enter limit \n");  
 scanf("%d",&n);  
   
 for( int i = 1 ; i<=n ; i++){  
   
 printf("%d \n",a);  
 c=a+b ;  
 a=b ;  
 b=c ;  
   
 }  
   
 }

1. Palindrome number : the reverse of a number is equal the number

5225

#include <stdio.h>  
int main() {  
 int n,r ,sum=0;  
 printf("Please enter a number you need to check if its plaindrome : \n");  
 scanf("%d",&n);  
 int temp = n;  
   
 while ( n>0){  
   
 r = n%10;  
 sum= sum\*10+r ;  
 n=n/10 ;  
 }   
 n=temp ;  
 if (n==sum)  
 {  
 printf("%d ,is a palindrome number ",n);  
   
 }  
 else{  
 printf("%d ,is not a palindrome number ",n);   
   
 }

}

1. Prime number or not: prime is a number which has 2 factors.

#include <stdio.h>  
int main() {  
 int n,i ,count=0;  
 printf("Please enter a number you need to check if its Prime number : \n");  
 scanf("%d",&n);  
   
 for(i=1;i<=n ;i++){  
 if (n%i==0){  
 count ++ ;  
 }  
   
 }  
 if (count ==2) {  
 printf("%d , is a Prime number ",n);  
   
 }  
   
 else {  
 printf("%d , is not a Prime number ",n);  
 }  
   
   
 }

1. Strong numbers :

Strong number mean the factorials each element in the number is equale the number itself

145

1!+4!+5!

1+24+120 = 145

#include <stdio.h>  
int main() {  
 int n,i ,r;  
 printf("Please enter a number you need to check if its Strong number : \n");  
 scanf("%d",&n);  
 int sum =0 ;  
 int temp =n ;  
 while (n>0){  
 r= n%10 ;  
 int fact=1;  
 for(i=r;i>=1 ;i--){  
 fact =fact\*i;  
 }  
 sum=sum+fact ;  
 n= n/10 ;  
 }  
   
 n=temp ;  
 if (n ==sum) {  
 printf("%d , is a strong number ",n);  
   
 }  
   
 else {  
 printf("%d , is not a strong number ",n);  
 }  
   
 }

1. Strong numbers in a giving range:

#include <stdio.h>  
int main() {  
 int limit,n,i ,r;  
 printf("Please enter a number you need to check if its Strong number : \n");  
 scanf("%d",&limit);  
   
   
   
 for(n=1;n<=limit;n++){  
 int temp =n;  
 int sum =0 ;  
 while (n>0){  
 r= n%10 ;  
 int fact=1;  
 for(i=r;i>=1 ;i--){  
 fact =fact\*i;  
 }  
 sum=sum+fact ;  
 n= n/10 ;  
 }  
 n=temp;  
 if (n ==sum) {  
 printf("%d , is a strong number ",n);  
   
 }  
   
 else {  
 }  
   
 }   
 }

1. Printing Perfect number in a range :

#include <stdio.h>  
int main() {  
 int limit ,n ,i ,sum;  
 printf("Please enter the limit to check the perfect numbers in it \n");  
 scanf("%d",&limit);  
   
 for(n=1;n<=limit ;n++){  
 int temp=n ;  
 sum=0 ;  
 for( int i = 1 ; i<n ; i++){  
   
 if (n%i==0){  
 sum =sum+i ;  
   
 }  
 n=temp;   
 }  
 if(n==sum){  
 printf("%d , is a perfect number.\n",n);  
   
 }   
 else{  
 //printf("%d , is not a perfect number.\n",n);  
   
 }   
   
}  
  
}

1. Armstrong Number:

The sum digit cubs is the same number

Example : 153

1^3+5^3+3^3

1+125+27

153

#include <stdio.h>  
int main() {  
 int n ,r ,c,sum=0;  
 printf("Please enter a number to check it is Armstrong or not : \n");  
 scanf("%d",&n);  
 int temp=n ;  
   
 while(n>0){  
   
 r=n%10 ;  
 c = r\*r\*r ;  
 sum=sum+c;  
 n=n/10 ;  
 }  
 n=temp ;  
 if(n==sum){  
   
 printf("%d , is an armstrong number ",n);  
   
 }  
 else{  
   
 printf("%d , is not an armstrong number ",n);  
   
 }  
 }

1. Armstrong Numbers in a range : 1, 153,370,371,407

#include <stdio.h>  
int main() {  
 int limit, n ,r ,c,sum;  
 printf("Please enter a limit to check all its Armstrong numbers : \n");  
 scanf("%d",&limit);  
 int temp=n ;  
 for(n=1 ;n<=limit ;n++){  
 int temp=n ;  
 sum=0;  
 while(n>0){  
   
 r=n%10 ;  
 c = r\*r\*r ;  
 sum=sum+c;  
 n=n/10 ;  
 }  
 n=temp ;  
 if(n==sum){  
   
 printf("%d , is an armstrong number.\n ",n);  
   
 }  
 else{  
   
 //printf("%d , is not an armstrong number ",n);  
   
 }  
 }  
 }

1. reverse Numbers of a given number:

#include <stdio.h>  
int main() {  
 int n ,r ,c,sum=0;  
 printf("Please enter the number you want to reverse it : \n");  
 scanf("%d",&n);  
 int temp=n ;  
   
 while(n>0){  
   
 r=n%10 ;  
   
 sum=sum\*10+r;  
 n=n/10 ;  
 }  
   
 n=temp ;  
   
 printf("%d , is the Reverse number of %d ",sum ,n);  
   
   
 }

1. find factorial using recursion method:

#include <stdio.h>  
int fact(int n);  
 int main() {  
 int n ;  
 printf("Please enter the number to find the factorial : \n");  
 scanf("%d",&n);  
   
   
 printf("result:%d", fact(n));  
 return 0 ;  
 }  
   
 int fact (int n)  
 {  
 int res ;  
 if (n==0){  
 res =1 ;  
 }  
 else{  
   
 res =n\*fact(n-1) ;  
   
 }  
 return res ;  
 }