**Factorial of a number**

#include<stdio.h>

int main()

{

int i,fact=1,number;

printf("Enter a number: ");

scanf("%d",&number);{

if(number<0){

printf("Please enter a positive integer...");

return 0;}

else

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

fact=fact\*i; }

}

printf("Factorial of %d is: %d",number,fact);

return 0;

}

**Perfect number**

#include<stdio.h>

#include<conio.h>

int main ()

{

int n,i,sum=0;

printf("enter n\n");

scanf("%d",&n);

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

{

if (n%i==0)

sum=sum+i;

}

if (n==sum)

printf("Entered no. is a perfect no. ");

else

printf("Entered no. is not a perfect no.");

getch();

return 0;

}

**Reversed number**

#include <stdio.h>

int main(){

int Num, rev\_Num = 0, remainder;

printf("Enter the number to reverse: ");

scanf("%d", &Num);

while (Num != 0){

remainder = Num % 10;

rev\_Num = rev\_Num \* 10 + remainder;

Num = Num/10;

}

printf("The reversed number is: %d", rev\_Num);

return 0;

}

**Arithmetic operators**

#include <stdio.h>

int main()

{

char operation;

double n1, n2;

printf("Enter an operator (+, -, \*, /): ");

scanf("%c", &operation);

printf("Enter two operands: ");

scanf("%lf %lf",&n1, &n2);

switch(operation)

{

case '+':

printf("%.1lf + %.1lf = %.1lf",n1, n2, n1+n2);

break;

case '-':

printf("%.1lf - %.1lf = %.1lf",n1, n2, n1-n2);

break;

case '\*':

printf("%.1lf \* %.1lf = %.1lf",n1, n2, n1\*n2);

break;

case '/':

printf("%.1lf / %.1lf = %.1lf",n1, n2, n1/n2);

break;

}

return 0;

}

**Add two matrices**

#include <stdio.h>

int main() {

int r, c, a[100][100], b[100][100], sum[100][100], i, j;

printf("Enter the number of rows (between 1 and 100): ");

scanf("%d", &r);

printf("Enter the number of columns (between 1 and 100): ");

scanf("%d", &c);

printf("\nEnter elements of 1st matrix:\n");

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

for (j = 0; j < c; ++j) {

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

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

}

printf("Enter elements of 2nd matrix:\n");

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

for (j = 0; j < c; ++j) {

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

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

}

// adding two matrices

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

for (j = 0; j < c; ++j) {

sum[i][j] = a[i][j] + b[i][j];

}

// printing the result

printf("\nSum of two matrices: \n");

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

for (j = 0; j < c; ++j) {

printf("%d ", sum[i][j]);

if (j == c - 1) {

printf("\n\n");

}

}

return 0;

}

**Add two numbers**

#include <stdio.h>

int main()

{

int first, second, \*p, \*q, sum;

printf("Enter two integers to add\n");

scanf("%d%d", &first, &second);

p = &first;

q = &second;

sum = \*p + \*q;

printf("Sum of the numbers = %d\n", sum);

return 0;

}

**Add complex numbers**

#include<stdio.h>

/\* Declaring Structure \*/

struct complex

{

float real;

float imaginary;

};

int main()

{

/\* Declaring structure variable using struct complex \*/

struct complex cnum1, cnum2, sum;

printf("Enter real and imaginary part of first complex number:\n");

scanf("%f%f", &cnum1.real, &cnum1.imaginary);

printf("Enter real and imaginary part of second complex number:\n");

scanf("%f%f", &cnum2.real, &cnum2.imaginary);

sum.real = cnum1.real + cnum2.real;

sum.imaginary = cnum1.imaginary + cnum2.imaginary;

printf("SUM = %0.2f + i %0.2f", sum.real, sum.imaginary);

return 0;

}

**Array of elements**

#include<stdio.h>

void main (){

int i,n,a[100],count=0;

printf("enter size:");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

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

if(a[i]==2){

continue;

}

else if(a[i]%2==0){

count++;

}

}

if(count>2){

}

printf("total composite number are: %d",count);

}

**Ascending order**

#include<stdio.h>

#include<string.h>

main(){

int i,j,n;

char str[100][100],s[100];

printf("Enter number of names :\n");

scanf("%d",&n);

printf("Enter names in any order:\n");

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

scanf("%s",str[i]);

}

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

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

if(strcmp(str[i],str[j])>0){

strcpy(s,str[i]);

strcpy(str[i],str[j]);

strcpy(str[j],s);

}

}

}

printf("\nThe sorted order of names are:\n");

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

printf("%s\n",str[i]);

}

}

**Binary search**

#include <stdio.h>

int main()

{

int i, low, high, mid, n, key, array[100];

printf("Enter number of elementsn");

scanf("%d",&n);

printf("Enter %d integersn", n);

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

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

printf("Enter value to findn");

scanf("%d", &key);

low = 0;

high = n - 1;

mid = (low+high)/2;

while (low <= high) {

if(array[mid] < key)

low = mid + 1;

else if (array[mid] == key) {

printf("%d found at location %d.n", key, mid+1);

break;

}

else

high = mid - 1;

mid = (low + high)/2;

}

if(low > high)

printf("Not found! %d isn't present in the list.n", key);

return 0;

**book details using structure**

#include<stdio.h>

struct book

{

char book\_name[30];

char author[30];

int book\_id;

float price;

};

int main()

{

struct book b; // Here b is a variable of structure book

printf("Welcome to DataFlair tutorials!\n\n");

printf("Enter the book name: ");

fgets(b.book\_name, 30, stdin);

printf("Enter the author name: ");

fgets(b.author, 30, stdin);

printf("Enter the book ID: ");

scanf("%d",&b.book\_id);

printf("Enter the book price: ");

scanf("%f",&b.price);

printf("\nThe details of the book are:\n\n");

printf("The book name is: ");

puts(b.book\_name);

printf("The author name is: ");

puts(b.author);

printf("The book ID is: %d\n\n",b.book\_id);

printf("The book price is: %0.2f\n",b.price);

return 0;

}

**Call by reference**

#include <stdio.h>

long addTwoNumbers(long \*, long \*);

int main()

{

long fno, sno, sum;

printf("\n\n Pointer : Add two numbers using call by reference:\n");

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

printf(" Input the first number : ");

scanf("%ld", &fno);

printf(" Input the second number : ");

scanf("%ld", &sno);

sum = addTwoNumbers(&fno, &sno);

printf(" The sum of %ld and %ld is %ld\n\n", fno, sno, sum);

return 0;

}

long addTwoNumbers(long \*n1, long \*n2)

{

long sum;

sum = \*n1 + \*n2;

return sum;

}

**Call by reference 3 numbers**

#include <stdio.h>

void swapNumbers(int \*x,int \*y,int \*z);

int main()

{

int e1,e2,e3;

printf("\n\n Pointer : Swap elements using call by reference :\n");

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

printf(" Input the value of 1st element : ");

scanf("%d",&e1);

printf(" Input the value of 2nd element : ");

scanf("%d",&e2);

printf(" Input the value of 3rd element : ");

scanf("%d",&e3);

printf("\n The value before swapping are :\n");

printf(" element 1 = %d\n element 2 = %d\n element 3 = %d\n",e1,e2,e3);

swapNumbers(&e1,&e2,&e3);

printf("\n The value after swapping are :\n");

printf(" element 1 = %d\n element 2 = %d\n element 3 = %d\n\n",e1,e2,e3);

return 0;

}

void swapNumbers(int \*x,int \*y,int \*z)

{

int tmp;

tmp=\*y;

\*y=\*x;

\*x=\*z;

\*z=tmp;

}

**Composite number array of element**

#include<stdio.h>

void main (){

int i,n,a[100],count=0;

printf("enter size:");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

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

if(a[i]==2){

continue;

}

else if(a[i]%2==0){

count++;

}

}

if(count>2){

}

printf("total composite number are: %d",count);

}

**Cube and square of a number**

#include<stdio.h>

int main()

{

int n,sq, cube;

printf("Enter the Number: ");

scanf("%d",&n);

sq=n\*n;

cube=n\*n\*n;

printf("\nThe Square of %d is %d ",n, sq);

printf(" \nThe Cube of %d is %d ",n, cube);

return 0;

}

**Display prime numbers using function**

#include <stdio.h>

int checkPrimeNumber(int n);

int main() {

int n1, n2, i, flag;

printf("Enter two positive integers: ");

scanf("%d %d", &n1, &n2);

if (n1 > n2) {

n1 = n1 + n2;

n2 = n1 - n2;

n1 = n1 - n2;

}

printf("Prime numbers between %d and %d are: ", n1, n2);

for (i = n1 + 1; i < n2; ++i) {

flag = checkPrimeNumber(i);

if (flag == 1) {

printf("%d ", i);

}

}

return 0;

}

int checkPrimeNumber(int n) {

int j, flag = 1;

for (j = 2; j <= n / 2; ++j) {

if (n % j == 0) {

flag = 0;

break;

}

}

return flag;

}

**Dynamic memory allocation**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int i,n;

float \*element;

printf("\n\n Pointer : Find the largest element using Dynamic Memory Allocation :\n");

printf(" Input total number of elements(1 to 100): ");

scanf("%d",&n);

element=(float\*)calloc(n,sizeof(float));

if(element==NULL)

{

printf(" No memory is allocated.");

exit(0);

}

printf("\n");

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

{

printf(" Number %d: ",i+1);

scanf("%f",element+i);

}

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

{

if(\*element<\*(element+i))

\*element=\*(element+i);

}

printf(" The Largest element is : %.2f \n\n",\*element);

return 0;

}

**elements in an array and print the elements**

#include <stdio.h>

int main()

{

int arr1[25], i,n;

printf("\n\n Pointer : Store and retrieve elements from an array :\n");

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

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

scanf("%d",&n);

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

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

{

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

scanf("%d",arr1+i);

}

printf(" The elements you entered are : \n");

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

{

printf(" element - %d : %d \n",i,\*(arr1+i));

}

return 0;

}

**Even or odd**

#include<stdio.h>

int main()

{

int num;

printf("enter the number :");

scanf("%d",&num);

if(num==0)

printf("input cannot be 0");

else if(num<=0)

printf("input cannot be negative");

else if(num%2==0)

printf("the given number is even");

else if(num%2!=0)

printf("the given number is odd");

}

**Length of the string**

#include<stdio.h>

int main() {

char str[20], \*pt;

int i = 0;

printf("Pointer Example Program : Find or Calculate Length of String \n");

printf("Enter Any string [below 20 chars] : ");

gets(str);

pt = str;

while (\*pt != '\0') {

i++;

pt++;

}

printf("Length of String : %d", i);

return 0;

}

**Matrix**

#include<stdio.h>

int main()

{

int a[20][20],b[20][20],c[20][20],m,n,i,j;

printf("Enter the size of matices:\n");

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

printf("Enter the value of A matrix:\n");

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

{

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

{

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

}

}

printf("Enter the value of B matrix:\n");

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

{

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

{

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

}

}

printf("Sum of two matrices:\n");

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

{

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

{

c[i][j]=a[i][j]+b[i][j];

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

}

printf("\n");

}

}

**Maximum and minimum number**

#include <stdio.h>

#include <conio.h>

int main()

{

int a[1000],i,n,min,max;

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

scanf("%d",&n);

printf("Enter elements in array : ");

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

{

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

}

min=max=a[0];

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

{

if(min>a[i])

min=a[i];

if(max<a[i])

max=a[i];

}

printf("minimum of array is : %d",min);

printf("\nmaximum of array is : %d",max);

return 0;

}

**Maximum and minimum number**

#include<stdio.h>

int main()

{

int no1,no2;

int \*ptr1,\*ptr2;

printf("Enter first number:\n");

scanf("%d",&no1);

printf("Enter second number:\n");

scanf("%d",&no2);

ptr1=&no1;/\*ptr1 stores address of no1\*/

ptr2=&no2;/\*ptr2 stores address of no2\*/

if(\*ptr1>\*ptr2)

{

printf("Maximum number is %d",\*ptr1);

}

else

{

printf("Maximum number is %d",\*ptr2);

}

return 0;

}

**Mean median mode**

#include<stdio.h>

main()

{

int i,j,a[20]={0},sum=0,n,t,b[20]={0},k=0,c=1,max=0,mode;

float x=0.0,y=0.0;

printf("\nEnter the limit\n");

scanf("%d",&n);

printf("Enter the set of numbers\n");

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

{

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

sum=sum+a[i];

}

x=(float)sum/(float)n;

printf("Mean\t= %f",x);

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;

}

}

}

if(n%2==0)

y=(float)(a[n/2]+a[(n-1)/2])/2;

else

y=a[(n-1)/2];

printf("\nMedian\t= %f",y);

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

{

mode=0;

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

{

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

{

mode++;

}

}

if((mode>max)&&(mode!=0))

{

k=0;

max=mode;

b[k]=a[i];

k++;

}

else if(mode==max)

{

b[k]=a[i];

k++;

}

}

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

{

if(a[i]==b[i])

c++;

}

if(c==n)

printf("\nThere is no mode");

else

{

printf("\nMode\t= ");

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

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

}

}

**Multiplication of complex numbers**

#include<stdio.h>

/\* Declaring Structure \*/

struct complex

{

float real;

float imaginary;

};

int main()

{

/\* Declaring structure variable using struct complex \*/

struct complex cnum1, cnum2, mul;

printf("Enter real and imaginary part of first complex number:\n");

scanf("%f%f", &cnum1.real, &cnum1.imaginary);

printf("Enter real and imaginary part of second complex number:\n");

scanf("%f%f", &cnum2.real, &cnum2.imaginary);

mul.real = cnum1.real \* cnum2.real - cnum1.imaginary \* cnum2.imaginary;

mul.imaginary = cnum1.real \* cnum2.imaginary + cnum2.real \* cnum1.imaginary;

printf("PRODUCT = %0.2f + i %0.2f", mul.real, mul.imaginary);

return 0;

}

**Mulitiplication of complex numbers**

#include<stdio.h>

/\* Declaring Structure \*/

struct complex

{

float real;

float imaginary;

};

int main()

{

/\* Declaring structure variable using struct complex \*/

struct complex cnum1, cnum2, mul;

printf("Enter real and imaginary part of first complex number:\n");

scanf("%f%f", &cnum1.real, &cnum1.imaginary);

printf("Enter real and imaginary part of second complex number:\n");

scanf("%f%f", &cnum2.real, &cnum2.imaginary);

mul.real = cnum1.real \* cnum2.real - cnum1.imaginary \* cnum2.imaginary;

mul.imaginary = cnum1.real \* cnum2.imaginary + cnum2.real \* cnum1.imaginary;

printf("PRODUCT = %0.2f + i %0.2f", mul.real, mul.imaginary);

return 0;

}

**Reversed array**

#include <stdio.h>

int main(){

int num, i, j, array1[50], array2[50];

printf("Enter no of elements in array\n");

scanf("%d", &num);

printf("Enter array elements\n");

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

scanf("%d", &array1[i]);

for (i = num - 1, j = 0; i >= 0; i--,j++)

array2[j] = array1[i];

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

array1[i] = array2[i];

printf("The reversed array:\n");

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

printf("%d\n", array1[i]);

return 0;

}

**Student details using ptr**

#include <stdio.h>

struct student

{

char name[30];

int roll;

float perc;

};

int main()

{

struct student std;

struct student \*ptr;

ptr= &std;

printf("Enter details of student :: \n");

printf("\nEnter Name of student :: ");

scanf("%s",ptr->name);

printf("\nEnter Roll No of student :: ");

scanf("%d",&ptr->roll);

printf("\nEnter Percentage of student :: ");

scanf("%f",&ptr->perc);

printf("\nEntered details of student are :: \n");

printf("\nName : %s \n\nRollNo: %d \n\nPercentage: %.02f\n\n",ptr->name,ptr->roll,ptr->perc);

return 0;

}

**Student details using structure**

#include <stdio.h>

#include <string.h>

void main()

{

int rl,phy,che,ca,total;

float per;

char nm[20],div[10];

printf("Input the Roll Number of the student :");

scanf("%d",&rl);

printf("Input the Name of the Student :");

scanf("%s",nm);

printf("Input the marks of Physics, Chemistry and Computer Application : ");

scanf("%d%d%d",&phy,&che,&ca);

total = phy+che+ca;

per = total/3.0;

if (per>=60)

strcpy(div,"First");

else

if (per<60&&per>=48)

strcpy(div,"Second");

else

if (per<48&&per>=36)

strcpy(div,"Pass");

else

strcpy(div,"Fail");

printf("\nRoll No : %d\nName of Student : %s\n",rl,nm);

printf("Marks in Physics : %d\nMarks in Chemistry : %d\nMarks in Computer Application : %d\n",phy,che,ca);

printf("Total Marks = %d\nPercentage = %5.2f\nDivision = %s\n",total,per,div);

}

**Student details using union**

#include <stdio.h>

#include <string.h>

union student

{

char name[20];

char subject[20];

float percentage;

};

int main()

{

union student record1;

union student record2;

strcpy(record1.name, "Raju");

strcpy(record1.subject, "Maths");

record1.percentage = 86.50;

printf("Union record1 values example\n");

printf(" Name : %s \n", record1.name);

printf(" Subject : %s \n", record1.subject);

printf(" Percentage : %f \n\n", record1.percentage);

printf("Union record2 values example\n");

strcpy(record2.name, "Mani");

printf(" Name : %s \n", record2.name);

strcpy(record2.subject, "Physics");

printf(" Subject : %s \n", record2.subject);

record2.percentage = 99.50;

printf(" Percentage : %f \n", record2.percentage);

return 0;

}

**Substraction of complex numbers**

#include<stdio.h>

struct complex

{

double real,imag;

};

int main()

{

struct complex x,y,c;

printf("enter the value of x and y for first complex number: ");

scanf("%lf%lf",&x.real, &x.imag);

printf("enter the value of x and y for second complex number: ");

scanf("%lf%lf",&y.real, &y.imag);

c.real=x.real-y.real;

c.imag=x.imag-y.imag;

printf("Subtraction of complex numbers: %.2lf%+.2lfi",c.real,c.imag);

return 0;

}

**Sum of natural numbers**

#include <stdio.h>

int addNumbers(int n);

int main() {

int num;

printf("Enter a positive integer: ");

scanf("%d", &num);

printf("Sum = %d", addNumbers(num));

return 0;

}

int addNumbers(int n) {

if (n != 0)

return n + addNumbers(n - 1);

else

return n;

}

**Simple interest**

#include<stdio.h>

int main()

{

int p,n;

char a[5];

printf("Enter the principal amount:");

scanf("%d",&p);

if(p<0)

{

printf("Enter valid amount.... Try again!!!");

}

else

{

printf("Enter no of years:");

scanf("%d",&n);

if(n<0)

{

printf("Enter valid no of years.... Try again!!!");

}

else

{

printf("Is customer senior citizen(y/n)?:");

scanf("%s",&a);

if (strcmp(a, "y") == 0)

{

printf("Intrest=%d",(p/100)\*n\*12);

}

else if(strcmp(a,"n")==0)

{

printf("Intrest=%d",(p/100)\*n\*10);

}

}

}

}