ARRAYS:

<u>Definition</u>: "Array is a collection of Homogeneous data item called an element of the array, in a continuous memory location with a single name."

- → The data type of the elements may be any valid data type like chan, into on float of estimates
- → The elements of the armay shares the same variable name but each element has a different index number known as subscript.
- → Armay can be single-dimensional on multidimensional. The number of subscripts determines the dimension of armay.
- A one-dimensional armay has one subscript, two dimensional armay has two subscripts and so on.
- → The one-dimensional arrays are known as vectors and two dimensional arrays are known as matrices.

One - Dimensional Armay:

Declaration of 4-D Annay: eNotes.in

The syntax for declaration of a 1-0 array is;

data-type array-name [size];

Here, array-name denotes the name of the array and it can be any valid c identifier, data-type is the data type of the elements of array. The size of the array specifies the number of elements that can be stoned in the array. It may be a positive integer constant on constant integer expression.

Example: int age [100];
float sal[15];
chan gnade[50]; etc.

Hene, age is an integer type array, which can stone 100 elements of integer type. The array sal is a floating type array of size 15, can hold Ploat values and grade is a chan type array, which can stone so elements of chan type.

→ The <u>symbolic constants</u> can be used to specify the array Size. For example;

clefrine Size 10

Void main()

{
 int mark[Size];

 float arm [Size];

→ The Use of symbolic constant to specify the Size of array makes it convenient to modify the program if the size of array is to be changed latter, because the size has to be changed only at one place i.e. in the #define directive.

Initialization of 1-D Annay:

- The general form of initializing an array is;

data-type annay-name [size] = { list of values};

-> Example: int num[4] = {101, 102, 103, 104};
[load a[3] = {4.15, 5.32, 1.19};

char name [4] = { '\n', '\t', '\n', '\f'};

- Armay can be initialized by listing the values in {} braces.
- If an array will be initialized at the time of the declaration the it will be of static kind.
- * values not initialized are set to zero.
- when an array is initialized, the dimensional value may be omitted. The number of elements in an array will be deter-mined by the compiler.

in a [8] = { 5,5,10,12,17 }

print ("1.d", a[2]); output = 10

print ("1.d", a[0]); owpul = 5

printf ("1.d", a[1]); output = 0 tes in
printf ('1.d", a); output = 100 (nddress of 1st element)

Name of the array holds the stanting address of the block where array elements are started.

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

printf ("1.d", a(i)); output: 55101211

Processing 1-D Armay:

- and the loop variable is used at the place of subscript.
- → The tinitial value of loop variable is taken 0. since arriary subscripts starts from 0.
- → The loop variable is increased by one each time, so that we can access and process the next element in the armay.
- The total no. of passes in the loop will be equal to the number of elements in the annay and in each pass we will process one element.
- -> suppose arrific is an arriay of int type -
 - (i) Reading values in ann[10]

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

 scanf ("1.d", & ann[i]);
 - (ii) <u>Displaying values</u> in aur(10]

 for (1=0; 1<10; (++) 1 es. 11)

 print ("1-d", aur(1));
 - In 1-D array address of the position of the element can be found out by the formula;

Example:
$$\alpha = 100$$
 $(100 + 2 \times (2))$ $\alpha = \frac{0}{1}$ $\frac{1}{2}$ $\frac{3}{3}$ address = $\frac{100}{100}$ $\frac{1}{100}$ $\frac{2}{100}$ $\frac{3}{100}$ $\frac{3}{100}$ $\frac{1}{100}$

```
/* write a program to input the values in an array & display it */
# include < stdio.h>
# include < conio.h)
Void main ()
   in aunsj, i : Notes in
    Clasca();
   for ( i=0; i<5; i++)
       } printf ( "Enter the elements for ann[1.d]: \n", i);
          scanf ("1.d", &arn[i]);
   praintf ( "The armay elements are : \n");
   for ( i=0; i<5; i++)
           print ("1d11", ann[i]);
   getch();
             LectureNotes.in
           Enter the elements for arm[o] : 2
  owpul:
           Enter the elements for armfiz: 5
           colon the elements for aun [2] 1:07 es in
            Enter the elements for arra[3] ; q
            Enter the elements for arm [47: 10
          The annay elements are:
```

```
1x write a program to find the sum of the elements in an arricy */
 # include (stdie.h)
 # include < conio.h>
  Void main ()
 closecu();
    for (i=0; i<5; i++)
           print ( " Enter the elements of an array: \n");
            scanf ( " 1.d", lann [i] );
              Sum = Sum + arca[i];
   printf ( " sum of elements in an array is : 1/d/n", sum);
   gelch();
 output: Enter the elements of an arriay:
            Gecture Notes.in
         Sum of elements of an array is: 47
1 * write a program to count the number of even and odd
     elements in an array */
# include < sidio.h)
# include < conic.h>
# define SIZE 5
void main()
   int i, ann[size], even = 0, odd = 0;
```

```
chaca();
for ( i = 0 ; i < sizE ; i++)
        printf ("Enter the elements of an array: \n");
         scanf (" /d", law[i]);
     if ( am [i] / 2 = = 0)
          Lecturence ; s.in
            else s
                   odd ++ ;
  printf (" even no.s = 1.d odd no.s = 1.d \n", even, odd);
  gelch();
output: Enten the elements of an arriay
              LectureNotes.in
        even no.s = 2 odd no.s = 3
/ * write a program to neverse the elements of an armay */
# include <stdio.h>
# include < Conio. h>
void main()
   inl i, j, arr [10] = {1, 2, 3, 5, 7, 9, 12, 4, 10, 51}, lemp;
   Chaca();
   for ( i=0 , j=9 ; t<j ; t++ , j--) ;
```

```
lemp = a[i];
    a[i] = a[i];
    a[i] = temp;
  printf ("The neverse elements of an array are : \n");
    for ( i=0; i<10; i++)
       | Print (" \d/t", arr (i));
 getch();
output: The neverse elements of an annay are:
          51 10 4 12 9 7 5 3 2 1
                        on
# include (stdio.h)
#include (conio.h)
 void main()
    in a [100], n, i;
    chaca(); ecture Notes.in
   proint ( " Enter the no. of numbers in an array : \n');
   scanf (" 1.d", 20);
   prontf (" Enter the numbers in an average
       for ( i=0; i<n; i++)
           scanf ( " /d", & a[i]);
  printle ( " The arrivery elements are : /n");
       for (i=0; i<n; (++)
            proof ( " 1d", a[i]);
```

```
printf ( " The array in neverse order is: \n");
   for ( i=n-1; i>=0; i--)
        prints ( " 1.4/4", a(1));
 gelch(); .
         Enter the no. of numbers in an array:
        LectureNotes.in
          Enter the numbers in an arrivery:
              9 10 12
            5
         The array elements are:
          The annay in neverse order is:
                  10 9 5
              12
  write a program to search an item in an arriay *
( Linear Search )
# include <sldio.h>
# include < conio-b) Cture Notes. 1
void main()
   inl i, item, aur [50], flag = 0, 51;
                               tureNotes.in
   chuch();
   pruntl ( " Enter the no of elements in an averay : \n");
   scanf (" 1.d", & n);
   prunts ( " Enter the elements of an array: \n");
        for ( i= 0; icn; i++)
           Scanf (" 1d", & ann (i]);
```

```
printf ( " Enter the item to be seanched : \n");
    scanf ("1d", & ilem);
  for ( i=0; i<n; i++)
          if ( item = = arrili)
                  prints ( " 1.d found at position 1.d/n", item, i+1)
       Lecture Plag = 1;
                  break;
 if ( flag = = 0)
      prantf ("item 1 id is not found in an armay: In", item);
  geich();
output: Enter the no. of elements in an array: 7
          Enter the elements of an array:
          Enter the item to be seanched: 3
          3 found at position 5
/ * write a program to find the maximum and minimum
    element in an armay * 1
# include < sidio h>
# include < conio.h)
 void main()
  int i, aver(10] = { 2.5.4.1.8.6.9,3.1,12}.
   int min, max;
```

```
min = max = arn[0];
for ( i= 0; i < 0q; i++)
           if ( arra[i] & min)
              - min = arm [i];
            if ( am[i] > max)
        Licture max = aun[i];
preint ( " Minimum = 11d Maximum = 1.d/n", min, max);
getch();
ow put: Minimum = 1 Maximum = 12
* write a program to sout the elements in an armay */
# include (sidio.h)
# include <conio.h>
void main()
int a (10], i,n, temp, j=0;
  cliucio; Lecture
  pruntle ( " Enter the no. of elements in an array: \n");
      scant ("1d", 8n);
   prints ( " Enter the elements to an annay : \n");
         for(i=0; i(n; (++)
             scans (" 1d", sa[i]);
   for ( i = 0; i< n = 0; i++)
         for ( ) = 1+1 ; j<n; j++)
                   if (afig > afig)
                         temp = alij :
```

```
a[i] = a[j];
           a [i] = temp;
      3
 praintf (" The sorted arriay elements are: \n");
         for( i=0; i<n; i++)
       Lecture prunis (" 1.d", asi]);
    getch();
         Enter the no. of elements in an arriay: 9
          Enter the elements to an array
              5 7 10 2 11 99 51 69
          The sorted array elements are:
              3 5 7 10 11 57 69 99
/* write a program to merge two arrays of no.s into a
   new annuge*dtureNotes.in
# include < stdio.h>
# include < conio.h>
void moun()
   in a.b.j. K.l;
   th and [10], and 2[10], and 3[20];
   (lower();
  pruntl ( " Enter the no of elements of armay 1 : In");
  scont (" .td", &a);
  provil ( " Enter the no of elements of armay. 2: /n");
```

```
Scanf ("1d", 8b);
printl ( " Enter 1.d elements of array 1 in sonted order : (n");
   for (j=0; j<a; j++)
       scanf (" ./.d", lawr1[i]);
printl ( " Enter 1.d elements of array 2 in souted order: \n");
    fon ( K= 0; K<b; K++)
        scarl ( " 1.d", 2 aur 2 [ k] );
 j = 0 ;
 K = 0;
 l = 0;
 while ( I < BL & R K < b)
            if ( wunt[j] < = ama[k])
                  ann 3(1) = ann 1[j];
                    j++;
           else
                am 3(1) = am 2[K];
                ektire No
                  (++)
 while ( ica)
                            Lecture Notes.in
           ann3[1] = ann1[j];
              1++;
               j++;
  while ( K < b)
           ann 3[1] = ann 2[k];
              1++;
              k++;
```

こうこうこうしゅ つりゅつ

```
printf (" New menged arriay in sorted order is: \n");
   for ( j=0; j<l; j++)
       print ( "1.d/+", aur3[j]);
   getch();
          Enter the no. of elements of array 1:4
           Enler the no. of elements of average : 4
          Enter 4 elements of armay 1 in sorted order:
            3
            4
           Enter 4 elements of away 2 in sorted order :
            9
            10
            20
        New menged array in sorted order is:
/ * write a program to search an item in an array using
    Binary search * 1
# include <sidio.h>
# include < conio. h>
void main()
   int articol, low = 0, high, mid, item, n;
   chaen();
  printl(" Enter the no of elements you want to insent in an
    scanf ( " 1.d", kon);
```

```
printf ( " Enter the elements in the array in sorted order: In);
 for ( i=0; (< n; i++)
       printf ("Enter an element: \n");
        scanf (" 1.d", 9 ali]);
printf ( " Enter the item to be searched : In");
   Scant ( " 1.d", & item);
   high = n-1 ;
   mid = (10w+high) | 2;
while ( low <= high && a[mid] ! = item)
         if ( a [ mid ] < item )
              low = mid+1;
               high = mid-1;
             mid = (low+ high) /2;
         printf (" Item is found at position 1.d \ n", mid+1);
  if ( a [mid] = = [tem)
  else praints (" Item is not found in an armay: \n");
                            Lecture Notes.in
 getch();
output: Enter the no. of elements you want to invent in an
        array: 5
       Enlen the elements in the annay in sorted orders:
         Enter an element: 39
         Enten an element: 45
```

Enlen an element: 59

Enter an element: 62

Enter an element: 99

Enter the item to be seanched: 59

Llem is found at position 3

Traversalt; reNotes.in

Definition: Traversal means visiting each elements in an array exactly once.

Myorithm: Traverse (a.n)

fon i ← 1 to n

Apply process to alij

End for

Here the process may be:

- 1) Displaying the Contents of an array.
- 2> Finding max m & min elements in an array
- 3> searching an element in an arriay

Lecture Notes.in

Two - Dimensional Annay:

- → A two dimensional array is an array of one dimensional arrays.
- A single dimensional array can stone a list of values. Whenever a table of values.

Declaration of 2-D Array:

The syntax for the declaration of a 2-D array is;

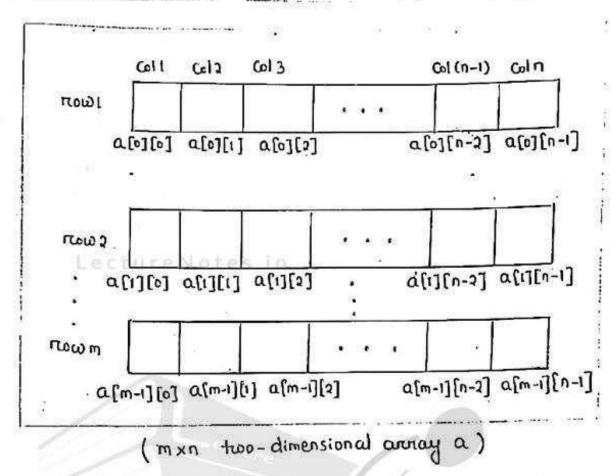
data-type array-name [Row-size][Column-size];

Here, row-size specifies the number of rows and column-Size specifies the number of columns in the array. The total no. of <u>elements</u> in the array are row-size * column-size.

Example: inla[4][5];

Here array is 'a' with 4 news and 5 columns. The individual elements of this array can be accessed by applying two subscripts, where the first subscript denotes the ruco number and the second subscript denotes the column number. The starting element of this arriay is afollol and the last element is a [3][4]. The total no of elements in this arriay is

→ In two-dimensional curray, the element declaration (both row and column) done with zero origin subscripting.



→ The array int a[3][3] will have elements as;

a[0][0] a[0][1] a[0][3]

a[1][0] .a[1][1] a[1][2]

a[2][0] a[2][1] -a[2][2] + e s in

→ A 2-D array can be called as a table, as they stones the table of values.

Initialization of 2-D Annay:

-> The general form of initializing a 2-0 array is;

dula-type array-name [now-size][column-size] =
f list of values };

-> Example: in a[3][3] = { 9.9.9.10,10,10,11,11};
on in a[3][3] = { {9.9.9}, {10.10,10}, {11,11,11}};

a (0)(0)	9
[0][1]	9
[0][2]	9
[1][0]	10
[ו][ו]	10
(1] [2]	10
[2][0]	11
[2][1]	ula
[2][2]	11

$$a = . \begin{bmatrix} 9 & 9 & 9 \\ 10 & 10 & 10 \\ 11 & 11 & 11 \end{bmatrix}_{3 \times 3}$$
otes.in

→ int a[4][3] = {11,10,12,9,8,7,6,13,15,19,21,62};

$$C = \begin{bmatrix} 11 & 10 & 12 \\ 9 & 9 & 7 \\ 6 & 13 & 15 \\ 19 & 21 & 62 \end{bmatrix}$$

$$4 \times 3$$

Lecture Notes.in

→ A 2-D annay can also be stoned like;

int a[3][3] = { 9, 10,52,45,6,5,3,13,15};

	<u>0 ယဘာ</u>			ποωι			news		
	610	coll	(612	Colo	11ع	(o) 2	دوان	Coll	Co1 2
L	9	10	52	45	6	5	3	13	15
	100	102	104	106	108	110	113	114	116

In 2-0 armay the address of the position of the element can be found out by the formula;

	Re	wo	1 100	ωı	Lows	
	610	Coll	Colo	Coll	COLD	110
a	5	7	٩	8	2	4
	100	102	104	106	108	110

Processing 2-D Annay:

- For processing 2-0 arrays, we use two nested for loops.
- The owner for loop connesponds to the row and the inner for loop cornesponds to the column.

int our [4](5];

(i) Reading values in array and;

(i) <u>Displaying</u> values of armay ann;

for (1=0; (<4; i++)

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

printl ("1d/l", arm[i][i]);

```
1 * write a program to input and display the matrix *1
# include (sidio.h)
# include < conio.h>
Void main ()
    in a [10][10], i, j, n.c;
    chuch();
 printf ( " Enter the order of matrix a: \n");
 scanf ("1.d1.d", &n, &c);
prints ( "Enter I'd elements to a matrix a: \n", x xc);
      fon(i=0; i<1; i++)
      for (j=0; j<c; j++)
          scanf (" 1.d", &alislis);
printf (" Elements of matrix a are: \n");
        fon (i=0; i<n; i++)
            { for ( )=0; i<c; j++)
            [ = ( printf ( " 1. d (t", a (i)(i));
               printf (" /n");
       getch();
                           Lecture Notes.in
        Enter the order of matrix a: 2
        Enler 6 elements to a matrix a:
```

```
Elements of matrix a are:
         3
         5
/ * Addition of two matrices */
# include < stdio.h>
# include < conio.h>
void main() ure Notes. in
  in a [10][10], b[10][10], d[10][10];
   inl i,j, π,c, p,q;
   chacuc);
 proint ( " Enten the order of matrix a: \n");
 Scanf (" 1.d 1.d", &n, &c);
 printf (" Enter 1.d values to matrix a: In", n *c);
      for ( i=0; ix n; i++)
      for (j=0; j<c; j++)
            scanf (" 1.d", &a(i)(i));
printl ( " Enlen the order of matrix b: \n");
  scant ( " 1.d 1.d", 2p, 2q,);
 printl ( " Enlen 1.d values to matrix b: \n" Pxq);
        for (i = 0; i<p; i++)
         for (j=0; j<q; j++)
              scanf ( " 1.d", & b[i][i]);
printf ( " Elements of matrix a: \n");
       fon ( i=0; (< n; i++)
```

```
for ( j=0; j<c; j++)
         preint ( " 1.6/t", a [i][i]);
     pruntf ( " \n");
  3
 prints ( " Elements of matrix b : 10");
     for ( i = 0 ; (< P ; i++)
          for ( j=0; j<q; j++)
               { pranif (" 1.d/t", b(i)[i]);
                3 printf ("10");
if (n = = p && c = = 9)
         for( (=0; i<n; i++)
                for ( j= 0; j<q; j++)
                   d(i)(i) = a(i)(i) + b(i)(i) ;
prints ( " The next ant matrix is : 10");
         fon ( i = 0; i < n; i++)
               for ( )=0; i<q; 1++) tes.in
                 printf (" 1.d/t", d[i][i]);
           prointf ( " \n") ;
      printl (" Addition is not possible : \n");
else
  getch();
```

1

```
out put :
      Enter the order of matrix a: 2
       2
       Enler 4 values to matrix a:
       Enter the order of matrix b: 2
    Le Sture Notes, in
       Enter 4 values to matrix 6:
       9
     Elements of matrix a:
        5
         4 3
     Elements of matrix b:
        9 8
        L'ectureNotes.in
    The nesultant matrix is:
          11 8
```

```
/* Malrix Mulliplication */
# include < sldio.h>
# include (conio.h)
void main()
  int a(10][10], b[10][10], ((10][10];
   inl i, j, p, q, m, n, K;
   Chaca(); cture Notes, in
 Print ( " Enter the order of matrix a: \n");
 scanf (" 1.d 1.d", 2m, 2n);
printly (" Enter the order of matrix b: \n");
 scanf ( " 1.d 1.d", 2p, 2q);
if (n!=P)
   printf ( " Matrix multiplication is not possible in");
  3
else {
       praintf ( " Enlen the elements of matrix a : \n");
          for (i=0; im; i+1) otes, in
            { for ( j = 0 ; j < n ; j + + )
               scanf ("1d", &a[i][i]);
      prints ( " Enter the elements of matrix b: \n");
            for ( i=0; i<p; i++)
                 for (j=0; j < q; j++)
                  scanf ("1.d", 26[1][1]);
prunif ( " Elements of malrix a : \n");
```

```
for ( i= 0; icm; i++ )
       for ( j=0; j<n; j++)
           { print ( " 1.d /t", ali][i]);
          printf ( " \n");
Prints ( " Elements of matrix b: \n");
     for ( i= 0; (< p; i++)
           for ( j=0; j<q; j++)
              printf (" 1.d/1", b[i][i]);
           prints ("\n");
prointf ( " The nexultant matrix c is: 10");
    for ( i = 0; ixm; i++ )
          for (j=0; j<q; j++)
Legture Notes.in
            for ( K = 0; K < P; K++)
               c[i][i] = (a[i][k] * b[k][i]) + c[i][i];
  for ( i=0; i<m; i++)
            for (j=0; j<q; j++)
              { prontf (" 1/-d", c [i][i]);
                }
buint( "10") ?
        gelch();
```

```
output: Enten the order of matrix a: 3
       3
       Enlen the order of matrix b: 3
       3
       Enlen the elements of matrix a:
       3
        LectureNotes.in
        4
       Enter the elements of mouthix b:
        2
        4
         7
         9
         12
       Elements of matrix ac Notes in
            8
              4 6
      Elements of matrix b:
                  5
            2
                12 10
            9
        nesultant matrix C 15:
                  120
         84
             101
         46
            ec.
                  88
             80 93
```

```
/* Transposing a matrix */ .
# include < stdio.h>
# include (conio.h)
void main()
   in a (10][10], i, j, π, c, b[10][10] j
   clusca();
 printf ( " Enter the order of matrix a: \n");
  scanf (" 1.d 1.d", &n. &c);
Printf (" Enler 1.d values to matrix a: \n", n*c);
     for ( i=0; ix n; i++)
     for (j=0; j<c; j++)
           Sconf ( " 1.d", &a[i7[i7]);
  printf (" Elements of matrix a are: \n");
        for ( i=0; i< n; i++)
             for (1=9;0) eq; (1+t)s.in
             prinif ("\n") ecture Notes.in
   for ( i=0; i< C; i++)
          for ( j = 0; j< n; j++)
                   b[i][i] = a[j][i] ;
  prints ( " After transposing matrix a is : \n");
```

```
for (i=0; i<c; i++)
      { for (j=0; j<n; j++)
           f praints ( " 1.2/t", b[i][i]);
      prints ("\n");
 gelch(); Lecture Notes. in
oulput: Enten the order of matrix a: 2
       Enter 6 values to matrix a:
        5
        Elements of matrix a are:
           Lesqure Notes.in
       After transposing matrix a is:
```

```
/ * write a program to check whether a matrix is symmetric
    on not */
#include <sidio.h>
# include (conio.h)
void main()
  int a[10][10], b[10][10], i, j, m, n
   chacaco; tureNotes.
 Print ( " Enter the order of matrix a: 10");
 scanf (" 1.d 1.d", &m, &n);
 Prunis ("Enter 1.d elements to matrix a: \n", m * n);
      for ( i=0; i<m; i++)
      for (j=0; j<n; j++)
             scant ( " 1.d", &a [i][i]);
prints (" Elements of matrix a arre: \n");
        for ( i = 0; i < m; i++)
             for ( j=0; i< m; j++ )
                   Print (" 1.dlt", alillis) ;
             prontf ("\n");
printf (" After transposing matrix a is: In");
       for ( i = 0; i< n; i++)
             for (i=0; i< m; j++)
                   ; [i][i]a = [i][i]d
```

```
for ( i= 0; i<n; i++)
   for ( i=0; j<m; j++)
        ; ([[i][i]d, "1/b.t") Itaing ?
    print ("10");
3
     LectureNotes.in
for ( i= 0; i<n; i++)
       for (j=0; j(m; j++)
            tf ( a[i][j] = = b[i][j])
                 Continue ;
                print (" Matrix is symmetric \n");
            3 6
          else
              printf ( " Not a symmetric meutrix");
       prints (" Matrix is Symmetric (n");
      Enlen the order of matrix a: 3
                        Lecture Notes, in
       Enter 9 elements to matrix a:
        5
```

```
Elements of matrix a are:
  After transposing matrix a is:
         5 6
      Lectute Notes.in
  Matrix is symmetric.
/* write a program to find out the sum of all the
    diagonal elements in a matrix *1
#include (sidio.h)
# include (conio.h)
void main()
3
        a [10](10], i, j, n.C, S;
    chacaci ecture Notes.in
    proints ( "Enter the order of matrix a: \n");
    scanf ( " 1.d 1.d", &n, &c);
   prints ( "Enter the elements of matrix a: (n");
        fon ( i= 0; i< n; i++)
             for ( j=0; i<c; 1++)
                 scanf ("1-d", 8 a[i][j]);
 prunts (" The elements of matrix a is: \n");
```

```
for ( i=0; i< n; i++)
      for (j=0; i<c; j++)
           } printf (" 1.d/t", a[i][i]);
        printf ("In");
S = 0;
if ( n = = c)
      fon ( i= 0; i< n; i++)
             for ( j=0; j< c ; j++)
          if(i+j==0 | 1| i+j==2 | 1| i+j==4)
1| i+j=6 | 1| i+j=8)
                s = s + a [i][i] ;
  praint ( " sum of diagonal = 1.d/n", s);
  3
 else
       prints ( " Addition of diagonal not possible (n");
  व्यक्ता;
output: Enter the order of matrix a: 3
          Enlen the elements of matrix a:
```

15

The elements of matrix a is:

2 5 8 10 12 50

Sum of diagonal = 77



LectureNotes.in

LectureNotes.in

```
Problems:
/ * write a program to search an element wing Binary search */
# include < stdio . h >
# include < conio.h)
void main()
٤
    tot a [90], low = 0, high, mid, n. item;
     clouder();
    printf (" Enter total no of element you want to insert in an array");
     scanf (" /.d", &n);
     high= n-1;
      mid = (low + high) /2;
 prints (" Enter the elements in souted order: (n");
   for ( i=0; isn; i++)
    scanf (" 1d', &a[i]);
 printf ( " Enter the item to be searched : In");
 Scanf (" 1.d", & item);
while ( low <= high && a[mid] != them) tes. 111
          if (a[mid] < ilem)
               1000 = mid+1;
                              Lecture Notes, in
              high = mid-1;
         mid = ( 10w + high) | 2;
if (a[mid] = = item)
        prunts (" Hem is found at position 1.d1n", mid+1),
else
       printf (" them is not found in an arracy");
     -golch();
```

OIP: Enter total no. of elements you want to insent in an curricy 7

Enter the elements in souled order :

3 9 12 15 29 44 59

Enler the item to be securched: 44

Item is found at position 6.

Selection Sout : Notes in

let us take a list of elements in unsorted order and surt them by applying selection sort.

Elements of the array are: -

40 20 50 60 30 10

pass 1: t=0

	j=1	j=2	j=3	j=4	j=5
0	40	20	26	20	20
1	20	40	40	40	40
ą	50	570	50	50	570
3	60	60	60	60	60
Ч	30	30	3c	30	30
5	10	10	lo	10	10
	EX	Total Control		1	EX

ans [0] > ans [1], Exchange

ans [0] > ans [2] No

ans [0] > ans [4] No

ans [0] > ans [5], Exchange

pare 2: ť≃L j=5 j=3 j=4 3=2 EΧ EX

an [1] > an [2] No an [1] > an [4], Exchange an [1] > an [4], Exchange

pau 3: (=2 j=5 j = 3j=4 EX EX

on [2] > arn [3] No

arn[2] > arn [4], Exchange

arn [2] > arn [5], Exchange

paus 4: ć=9 4=1 j=5

our [3] > and [4], Exchange

EX

Page 5: t=4 j=5

o to
1 ao
2 30

Lecture Notes.in

In pase 5 ann [4] > arrs [5], Exchange

Finally after pase 5 we got the souted array.

10 20 30 40 50 60

```
/* program of sorting wing selection soul */
# include (sidio.h)
# include (conio.h)
# define size 10
void main()
{
   th wur[SIZE];
   inl i,j, temp;
    chacu();
 printf ( " Enter elements of the array : \n");
    for ( i=0; ( < SIZE ; i++)
        Scanf ( " /d", Larrli]);
 for( i=0; i < size-1; i++)
      for (j=i+1; j < sizE; j++)
          f ( ( arr(i] > arr(i])
                                    otes.in
                    : [i] nu = [i] nu
                    arra[i] = temp;
                 3
            7
   printf ( " The sorted array is : \n" );
      for ( i=0; i < SIZE; i++)
           pronif ( " 1.d', ara[i]);
       pruntf ("\n");
 getch();
3
```

Bubble sont: let us take a list of element in unsonted order and sont them by using bubble sont.

Elements of the annay are -

40 20 50 60 30 10

paul: 2=0

	0 = L	N 1=1.5.	n j= 2	j=3	j=4
0	40	20	20	20	20
ι	20	40=	40	Чо	40
3	50	50	50	20	50
3	60	60	60	60	30
4	3t	30	30	30	60
5	to	10	10	10	16
	EX			EX	EX

on (2] > and (3] No

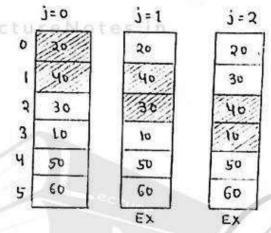
arn [3] > arn [4], Exchange

paux ? : ¿=1

	<i>j=</i> c	7=1 ec	tuję ₂ Not	j=3
O	20	20	20	20
1	40	40	40	Чо
2	50	50	50	30
3	30	30	30	50
Ч	10	10	10	10
5	60	60	60	€0
*		#6	EX	EX

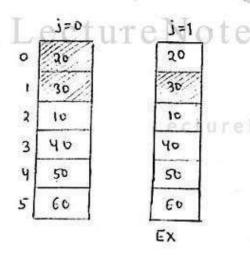
and [3] > and [4], Exchange and [1] > and [3] No and [1] > and [3] No

pas 3: ====



our [1] > our [2], Exchange our [2] > our [3], Exchange

paces 4: i=3



our[1] > our[2], Exchange

```
pass 5: t=4 j=0

0 20

1 10

2 30

3 40

4 50

Ex
```

```
In pall 5 arrs[0] > arrs[1], Excharge
Finally after poer 5 we got the sorted arriver.
       10 20 30 40 50 60
/* program of sorting wing bubble sort */
#include (stdio.h)
# include (cenio.h)
# define size 20
void main ()
  in amisize cture Notes. in
   int i, i, temp;
   clasen();
 printle ( " Enter elements of the array : (n");
    fon ( i = 0 ; i < SIZE ; i++ )
       sconf ("1.d", 8 wali]);
    for ( i=0 ; i < SIZE-L ; i++ )
          for (j=0; j< SIZE-1-i; j++)
```

```
if ( arr[i] > arr[i+1])
             temp = ann [i];
            ann[j] = ann[j+1];
            ant[j+1] = temp;
        "The sonted armay is : \n");
     for (i=0; ixsize; i++)
       printf ("1.d", arr(i));
    printf ("In");
 getch();
/* write a program to print the pascal triangle */
#include <sldio.h>
# include < conio.h>
# define SIZE 15
void main() Lecture Notes . i
٤
   int a[SIZE][SIZE];
    ini t,j,n;
    chacic);
  printf (" Enter n:");
   scont (".1.d", 20);
   For ( i=0 ; ik=n ; i++ )
        for ( j=0 ; j <= i ; j++ )
           · t[( j==0 11 z== j)
                a[i][i] = [;
```

```
else
    ac(i)(i) = a(i-i)(i-i) + a(i-i)(i);
  for (i=0; i=n; i++)
    { fon (j=0; j<=i; j++)
                    (" 1.d", a(i)(i));
get-ch();
                    Enter n: 7
output:
                     10 10
                         20 15 6
```

LectureNotes.in

Lecture Notes.in