ARRAYS

DELHI 2008

3.a) Write a function in C++, which accepts an integer array and its size as parameters and rearranges the array in reverse.

Example:If an array of nine elements initially contains the elements as 4, 2, 5, 1, 6, 7, 8, 12, 10
Then the function should rearrange the array as 10,12, 8, 7, 6, 1, 5, 2, 4

Solution:

```
void receive(int A[ ], int size)
{  int temp;
     for(i=0,j=size-1;i<size/2;i++,j--)
     {  temp=A[i];
        A[i]=A[j];
        A[j]=temp;
      }
 }//end of receive function.
```

3.b) An array Arr[40][10] is store in the memory along the column with each element occupying 4 bytes. Find out the base address of the location Arr[3][6] if the location Arr[30][10] is stored at the address 9000.

Solution: Children, Try this answer as an assignment.

3.d) Write a function in C++ to print the product of each column of a two dimensional array passed as the arguments of the function.

Example : If the two dimensional array contains

1	2	4
3	5	6
4	3	2
2	1	5

Then the output should appear as:

Product of Column 1 = 24
Product of Column 2 = 30
Product of Column 3 =240

```
void receive(int A[ ][ ],int r,int c)
{  int i,j,B[c];
   for(i=0;i<c;i++)
   B[i]=1;
   for(i=0;i<r;i++)
      for(j=0;j<c;j++)
           B[j]=B[j]*A[i][j];
   for(i=0;i<c;i++)
          cout<<"\nProduct of Column "<<i+1<<" = "<<B[i];
 }
```

OUTSIDE DELHI 2008

3.a) Write a function in C++, which accepts an integer array and its size as arguments and swap the elements of every even location with its following odd location.

Example : If an array of nine elements initially contains the elements as 2,4,1,6,5,7,9,23,10
then the function should rearrange the array as 4,2,6,1,7,5,23,9,10

```
void SwapArray(int A[ ], int N)
{   int i,j,temp;
   /*  cout<<"\nThe elements before doing the desired alterations…";
       for(i=0;i<N;i++)
              cout<<A[i]<<'\t';  */
   for(i=0;i<N-1;i+=2)
   {   temp=A[i];
       A[i]=A[i+1];
       A[i+1]=temp;
   }
  /*  cout<<"\nThe elements after completed the desired alterations…";
       for(i=0;i<N;i++)             cout<<A[i]<<'\t';  */
}
```

3.b) An array Arr[50][10] is store in the memory along the row with each element occupying 2 bytes. Find out the Base address of the location Arr[20][50], if the location Arr[10][25] is stored at the address 10000.

Solution: Children, Try this answer as an assignment.

3.d) Write a function in C++ to print the product of each row of a two dimensional array passed as the arguments of the function

Example: if the two imensional array contains

20	40	10
40	50	30
60	30	20
40	20	30

Then the output should appear as:

Product of Row 1 = 8000
Product of Row 2 = 6000
Product of Row 3 =3600
Product of Row 4 = 2400

```
void receive(int A[ ][ ],int r,int c)
{   int i,j,B[r];
    for(i=0;i<r;i++)
    B[i]=1;
    for(i=0;i<r;i++)
       for(j=0;j<c;j++)
            B[i]=B[i]*A[i][j];
    for(i=0;i<r;i++)
    cout<<"\nProduct of Row "<<i+1<<" = "<<B[i];
  }
```

DELHI 2007

3.a) Write function in C++ which accepts an integer array and size as arguments and replaces elements having odd values with thrice its value and elements having even values with twice its value.

Example : if an array of five elements initially contains elements as 3, 4, 5, 16, 9
The the function should rearrange the content of the array as 9, 8, 75, 32, 27

Solution:

```
void manipulate (int a[ ],int size)
{ for (i=0;i<size;i++)
  {    if (a[i]%2= =1)
       a[i]=a[i]*3;
       else
       a[i]=a[i]*2;
       cout<<a[i]<<',';
  }
}
```

3.b) An array Array[20][15] is stored in the memory along the **column** with each element occupying 8 bytes. Find out the base address of the element Array[2][3] if the element Array[4][5] is stored at the address 1000.

Solution:

Given Data: Aray [20][15] W=8 B=? R=20 C=15 $L_r = 0$ $L_c = 0$

Address of Array [2][3] =?

Address of Array[4][5] =1000.

Address of an element (I,J) in column major =B + W ((I-L_r) + R(J-L_c))

Therefore 1000=B+8*((4-0)+20(5-0))
1000=B+8*(4+20*5)
1000 =B+8*104
1000=B+832
B =1000-832
B =168

Therefore Address of Array[2][3]=168+8*((2-0)+20(3-0))
=168+8*(2+20*3)
=168+8*62
=168+496
=664

3.d) Write a function in C++ which accepts a 2D array of integers and its size as arguments and displays the elements which lie on diagonals. [Assuming the 2D Array to be a square matrix with odd dimension i.e., 3x3, 5x5 ,7x7 etc…]

Example : if the array content is

5 4 **3**
6 **7** 8
1 2 **9**

Out put through the function should be :

Diagonal One : 5 7 9
Diagonal Two : 3 7 1

Solution:

```
void accept(int a[ ][ ],int size)
{  cout<<"Diagonal One:";
   for (int i=0;i<size;i++)
         for(int j=0;j<size;j++)
               if (i= = j)
                  cout<<a[i][j]<<'\t';
   cout<<"\n Diagonal Two:";
   for (i=0;i<size;i++)
         for(j=0;j<size;j++)
               if((i+j)= =(size-1))
                  cout<<a[i][j]<<'\t';
}
```

OUTSIDE DELHI 2007

3.a) Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having even values with its half and elements having odd values with twice its value .

Example : If an array of five elements initially contains the elements as 3, 4, 5, 16, 9
then the function should rearrange content of the array as 6, 2, 10, 8, 18

Solution:

```
void accept(int a[ ],int size)
{  for (int i=0;i<size;i++)
   {    if (a[i]%2= =0)
           a[i]=a[i]/2;
        else
                a[i]=a[i]*2;
        cout<<a[i]<<',';
   }
}
```

3.b) An array Arr[15][20] is stored in the memory along the **row** with each element occupying 4 bytes. Find out the Base address of the location Arr[3][2], if the location Arr[5][2] is stored at the address 1500.

Solution: Given Data: Arr[15][20] W=4 B=? R=15 C=20 $L_r = 0$ $L_c = 0$

Address of Arr[3][2] = ?

Address of Arr[5][2] = 1500.

Address of an element (I,J) in row major = B+W(C(I-L_r)+(J-L_c))

Therefore, 1500 = B+4(20(5-0)+(2-0))

1500 = B+4(20*5+2)

1500 = B+4*102

1500 = B+408

B =1500-408

B=1092

Address of Arr[3][2] =1092+4(20*3+2)

=1092+4(62)

=1092+248 =1340.

3.d) Write a function in C++ which accepts a 2D array of integers and its size as arguments and displays the elements of middle row and the elements of middle column. [Assuming the 2D Array to be a square matrix with odd dimension i.e., 3x3, 5x5, 7x7 etc…]

Example : If the array content is

```
3  5  4
7  6  9
2  1  8
```

Output through the function should be :

Middle Row : 7 6 9

Middle Column : 5 6 1

Solution:

```
void accept(int a[ ][ ],int size)
{  cout<<"Middle Row:";
   for (int i=0;i<size;i++)
         for(int j=0;j<size;j++)
            if (i= = size/2)
               cout<<a[i][j]<<'\t';
   cout<<"\n Middle Column:";
   for (i=0;i<size;i++)
         for(j=0;j<size;j++)
            if(j= =size/2)
```

```
        cout<<a[i][j]<<'\t';
    }
```

DELHI 2006

3.a) Write function in C++ which accepts an integer array and size as arguments and assign values into a 2D array of integers in the following format :

If the array is 1, 2, 3, 4, 5, 6

The resultant 2D array is given below

1	2	3	4	5	6
1	2	3	4	5	0
1	2	3	4	0	0
1	2	3	0	0	0
1	2	0	0	0	0
1	0	0	0	0	0

If the array is 1, 2, 3

The resultant 2D array is given :

1	2	3
1	2	0
1	0	0

Solution:

```
void input (int a[ ],int size)
{  int b[size] [size];
   for (int i=0;i.<size;i++)
   {
     for (int j=0;j<size;j++)
      {
        if(( i+j)>=size)
              b[i][j]=0;
        else
              b[i][j]=a[j];
        cout<<b[i][j]<<'\t';
      }
     cout<<endl;
   }
}
```

3.b) An array MAT[30][10] is stored in the memory along column wise with each element occupying 8 bytes of the memory. Find out the Base address and the address of element MAT[20][5] , if the location MAT[3][7] is stored at the address 1000.

Solution: Children, Try this answer as an assignment.

OUTSIDE DELHI 2006

3.a) Write function in C++ which accepts an integer array and size as arguments and assign values into a 2D array of integers in the following format :

If the array is 1, 2, 3, 4, 5, 6

The resultant 2D array is given below :

1	0	0	0	0	0
1	2	0	0	0	0
1	2	3	0	0	0
1	2	3	4	0	0
1	2	3	4	5	0
1	2	3	4	5	6

If the array is 1, 2, 3

The resultant 2D array is given :

1	0	0
1	2	0
1	2	3

Solution:

```
void input (int a[ ],int size)
{  int b[size] [size];
   for (int i=0;i.<size;i++)
   {  for (int j=0;j<size;j++)
      { if(( i<j)
              b[i][j]=0;
        else
              b[i][j]=a[j];
        cout<<b[i][j]<<'\t';
      }
     cout<<endl;
```

```
        }
    }
```

3.b) An array MAT[20][10] is stored in the memory along the row with each element occupying 4 bytes of the memory. Find out the Base address and the address of element MAT[10][5] , if the location MAT[3][7] is stored at the address 1000.

Solution: Children, Try this answer as an assignment.

DELHI 2005

3.a) Write a function in C++ which accepts an integer array and its size as arguments and exchanges the values of first half side elements with the second half side elements of the array.

Example :

If an array of 8 elements initial content as 2, 4, 1, 6, 7, 9, 23, 10
The function should rearrange array as 7, 9, 23, 10, 2, 4, 1, 6

Solution:

```
void change(int a[ ],int size)
{
  int i,j,temp;
  for(i=0,j=size/2;j<size;i++,j++)
  {  temp=a[i];
     a[i]=a[j];
     a[j]=temp;
  }
}
```

3.b) An array Arr[15][35] is stored in the memory along the row with each of its element occupying 4 bytes . Find out the Base address and the address of element Arr[2][5] , if the location Arr[5][10] is stored at the address 4000.

Solution: Children, Try this answer as an assignment.

3.d) Write a function in C++ to print sum of all values which either are divisible by 2 or divisible by 3 present in a 2D array passed as the argument of the function.

Solution:

```
void Sum(int A[ ][ ],int R,int C)
{   int i,j,S=0;
    for(i=0;i<R;i++)
        for(j=0;j<C;j++)
            if(A[i][j]%2= = 0 ||A[i][j]%3= = 0)
                S=S+A[i][j];
    cout<<"\nThe Sum of all the values which are divisible by 2 or 3 in the array = "<<S;
}
```

OUTSIDE DEHI 2005

3.a) Write a function in C++ which accepts an integer array and its size as arguments and exchanges the values of first half side elements with the second half side elements of the array.

Example :

If an array of 8 elements initial content as 8, 10, 1, 3, 17, 90, 13, 60
The function should rearrange array as 17, 90, 13, 60, 8, 10, 1, 3

Solution: Refer Delhi 2005 Q.3a.

3.b) An array Arr[35][15] is stored in the memory along the row with each of its element occupying 4 bytes . Find out the Base address and the address of element Arr[20][5] , if the location Arr[2][2] is stored at the address 3000.

Solution: Children, Try this answer as an assignment.

3.d) Write a function in C++ to print sum of all values which either are divisible by 3 or divisible by 5 present in a 2D array passed as the argument of the function.

Ans:-

```
void Sum(int A[ ][ ],int R,int C)
{  int S=0,i,j;
   for(i=0;i<R;i++)
       for(j=0;j<C;j++)
           if((a[i][j]%3= =0)||(a[i][j]%5= =0))
               S=S+A[i][j];
   cout<<" nThe Sum of all the values which are divisible by 3 or 5 in the array = "<<S;
}
```

DELHI 2004

3.a) Define the function **SwapArray(int[ ], int),**that would expect a 1D integer array NUMBERS and its size N. the function should rearrange the array in such a way that the values of that locations of the array are exchanged. (Assume the size of the array to be even).

Example :

If the array initially contains {2, 5, 9, 14, 17, 8, 19, 16}

Then after rearrangement the array should contain {5, 2, 14, 9, 8, 17, 16, 19}

Solution:

```
void SwapArray(int NUMBERS[ ], int N)
{   int i,j,temp;
    /*  cout<<"\nThe elements before doing the desired alterations…";
        for(i=0;i<N;i++)
                cout<<NUMBERS[i]<<'\t';  */
    for(i=0;i<N-1;i+=2)
    {   temp=NUMBERS[i];
        NUMBERS[i]=NUMBERS[i+1];
        NUMBERS[i+1]=temp;
    }
   /*  cout<<"\nThe elements after completed the desired alterations…";
       for(i=0;i<N;i++)
               cout<<NUMBERS[i]<<'\t';  */
}
```

3.b) An array ARR[5][5] is stored in the memory with each element occupying 3 bytes of space. Assuming the base address of ARR to be 1500, compute the address of ARR[2][4], when the array is stored :

Solution: Children, Try this answer as an assignment.

3.c) Write a function in C++ to find the sum of diagonal elements from a 2D array of type float. Use the array and its size as parameters with float as its return type.

Solution:

```
float diasum(float A[ ][ ],int R,int C)
{  int i,j;
   float Dsum=0.0;
   for(i=0;i<R;i++)
       for(j=0;j<C;j++)
           if((i= = j)| | (i+j)= =(size-1))
                Dsum=Dsum+A[i][j];
  return Dsum;
}
```

DELHI 2003

3.a) Assume a array E containing elements of structure Employee is required to be arranged in descending order of Salary. Write a C++ function to arrange same with the help of bubble sort, the array and its size is required to be passed as parameters to the function. Definition of structrure Employee is as follows:

```
Struct Employee
{
    int Eno;
    char name[25];
    float Salary;
};
```

Solution:

```
void bubble(Employee E[ ],int n)
{ int i,j;
  Employee Etemp;
  for(i=0;i<n;++i)
     for(j=0;j<(n-1)-i ;j++)
           if(E[j].salary<E[j+1].salary)
            {              Etemp=E[j];
                           E[j]=E[j+1];
                           E[j+1]=temp;
             }
  cout<<"The details of the employee in ascending order of salary ";
  for(i=0;i<n;i++)
     cout<<E[i].Eno<<'\t'<<E[i].name<<'\t<<E[i].Salary<<endl;
 }
```

3.b) An array X[30][10] is stored in the memory with each element requiring 4 bytes storage. Find out the Base address of X is 4500, find out memory locations of X[12][8] and X[2][14], if the content is stored along the row.

Solution: Children, Try this answer as an assignment.

3.c) Write a user-defined function in C++ to display those elements of 2D array T[4][4] which are divisible by 100. Assume the content of the array is already present and the function prototype is as follows: void showhundred(int T[4][4]);

```
void showhundred(int T[4][4])
{   int i,j;
    cout<<"\nThe elements in the array which are divisible by 100 …..";
    for(i=0;i<4;i++)
        for(j=0;j<4;j++)
           if(T[i][j]%100= =0)
                cout<<T[i][j]<<'\t';
}
```

DELHI 2002

3.a) Define array and pointer.

Solution: An array refer to a named list of a finite number n of similar data elements. Each of the data elements can be referenced respectively by a set of consecutive numbers.

Arrays can be one dimensional, two dimensional or multi dimensional.

An array can be declared as : Syntax: data_type Array_name[size];

Eg: int A[10]; //Then location of the array are A[0], A[1],…….A[9].

int B[5][4]; //This array can holds 5 X 4 = 20 elements.

3.d) The array A[20][10] is stored in the memory with each element requiring one byte of storage if the base address of a is 0, determine the location of A[10][5] when the array A is stored by column major.

Solution: Children, Try this answer as an assignment.

3.c) Considering the following key set: 42,29,74,11,65,58, use insertion sort to sort the data in ascending order and indicate the sequences of steps required.

Solution:

In this, Suppose an array A with n elements A[1],A[2],…A[N] is in memory. The insertion sort algorithm scans A from A[1] to A[N], insertion each element A[K] into its proper position in the previously sorted subarray A[1],A[2],…,A[K-1].

This sorting algorithm is frequently used when n is small.

The array contains 6 elements as follows: 42,29,74,11,65,58

Pass	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]	A[6]
K=1	-32768	**42**	29	74	11	65	58
K=2	-32768	42	**29**	74	11	65	58
K=3	-32768	29	42	**74**	11	65	58
K=4	-32768	29	42	74	**11**	65	58
K=5	-32768	11	29	42	74	**65**	58
K=6	-32768	11	29	42	65	74	**58**
Sorted	**-32768**	**11**	**29**	**42**	**58**	**65**	**74**

DELHI 2001

3.a) Given two arrays of integers X and Y of sizes m and n respectively. Write a function named MERGE() which will third array named Z, such that the following sequence is followed.

(i) All odd numbers of X from left to right are copied into Z from left to right.
(ii) All even numbers of X from left to right are copied into Z from right to left.
(iii) All odd numbers of Y from left to right are copied into Z from left to right.
(iv) All even numbers of Y from left to right are copied into Z from right to left.

X, Y and Z are passed as arguments to MERGE().

Eg. X is {3, 2, 1, 7, 6, 3} and {9, 3, 5, 6, 2, 8, 10}

the resultant array Z is {3, 1, 7, 3, 9, 3, 5, 10, 8, 2, 6, 6, 2}

```
void MERGE(int X[ ], int m,int Y[ ],int n,int Z[ ])
{   int mn,i,,left=0,right=mn-1;
    mn=m+n;
    for(i=0;i<m;i++)
        if (X[i]%2= = 1)
          Z[left++]=X[i]; //For copying odd numbers of X into Z from left to right
        else
          Z[right- -]=X[i]; //For copying even number of X into Z from right to left
   for(i=0;i<n;i++)
        if (X[i]%2= = 1)
```

```
            Z[left++]=Y[i];   //For copying odd numbers of Y into Z from left to right
        else
            Z[right- -]=Y[i]; //For copying even number of X into Z from right to left
}
```

3.b) An array X[10][20] is stored in the memory with each element requiring 4 bytes of storage. If the Base address of the array is 1000, calculate location of X[5][15] when the array X is stored using column major order.

NOTE: X[10][20] means valid row indices are 0 and 9 and valid column indices are 0 and 19

Solution: Children, Try this answer as an assignment.

3.c) Write a user-defined function named Lower_half() which takes 2D array A, with size N rows and N columns as argument and prints the lower half of the array.

Eg.	2 3 1 5 0		2
	7 1 5 3 1		7 1
Input	2 5 7 8 1	the output will be	2 5 7
	0 1 5 0 1		0 1 5 0
	3 4 9 1 5		3 4 9 1 5

Solution:

```
void Lower_half( int A[ ][ ],int N)
{   int i,j;
    for(i=0;i<N;i++)
        for(j=0;j<N;j++)
        {   if(i<j)
                cout<<A[i][j]<<'\t';
            cout<<endl;
        }
}
```

DELHI 2000

3.a) Suppose A, B, C are arrays of integers of size M, N and M+N respectively. The numbers in array A appear in ascending order while numbers in array in descending order. Write user defined function in C++ to produce third array C by merging array A by B in ascending order. Use A, B and C as arguments in the function.

```
void Merge(int A[ ],int M,int B[ ],int N,int C[ ])
{   int a,b,c;
    for(a=0,b=N-1,c=0;a<M&&b>=0;)
    {       if(A[a]<=B[b])
                C[c++]=A[a++];
            else
                C[c++]=B[b--];
    }
    if(a<M)
    {       while(a<M)
                C[c++]=A[a++];
    }
    else
    {       while(b>=0)
                C[c++]=B[b--];
    }
}
```

3.b) An array VAL[1…15][1…10] is stored in the memory with each element requiring 4 bytes of storage. If the base address of the array VAL is 1500, determine the location of VAL[12][9] when the array VAL is stored (i) Row wise (ii) Column wise.

Solution:

Given Data:

VAL[1…15][1…10]
Word Length (W) = 4 Bytes
Base Address of VAL(B) = 1500
VAL[12][9] = ?
C = Total No of Columns   R = Total No of Rows
L_r = Least Row=1 L_c = Least Column=1

(i) Row Major:

Address of an element (I,J) in row major = B + W (C (I-L_r) + (J – L_c))

VAL [12][9] = 1500 + 4 (10 * (12-1) + (9-1))

= 1500 + 4 (10 * 11+8)
= 1500 + 4 (118)
= 1500 + 472
= 1972.

(i) Column Major:

Address of an element (I,J) in column major = B + W ((I-L_r) + R(J – L_c))

VAL [12][9] = 1500 + 4 ((12-1) +15 * (9-1))
= 1500 + 4 (11 + 15 * 8)
= 1500 + 4 (11+ 120)
= 1500 + 4 * 131
= 1500 + 524
= 2024.

3.c) Write a user-defined function in C++ to find and display the sum of diagonal elements from a 2D array MATRIX[6][6] containing integers.

```
void displaysum( )
{  int i,j,D1=0,D2=0,MATRIX[6][6];
   cout<<"\nEnter any 36 values….";
   for(i=0;i<6;i++)
       for(j=0;j<6;j++)
       {  cin>>MATRIX[i][j];
          if(i= = j)
              D1=D1+MATRIX[i][j];
          else if ((i+j)= =(size-1))
              D2=D2+MATRIX[i][j];
       }
    cout<<"\nThe sum of the elements of the Main Diagonal = "<<D1;
    cout<<"\nThe sum of the elements of the Other Diagonal = "<<D2;
}
```

DELHI 1999

3.a) Suppose a 1D array AR containing integers is arranged in ascending order. Write a user defined function in C++ to search for one integer from AR with the help of binary search method, to show presence of the number in the array. The function should have three parameters: (1) an array AR (2) the number to be searched and (3) the number of elements N in the array.

```
void BinSearch(int AR[ ], int Sno, int N)
{    int  l=0,u=N-1,m,flag=0;
     while(l<=u)
     {    m=(l+u)/2;
          if (Sno= = AR[m])
           {     flag=1;
                 break;
           }
          else if(Sno<AR[m])
                 u=m-1;
          else
                 l=m+1;
     }
     if( flag = = 0)
            cout<<"\nThe Search Element  "<<Sno<<" is not available";
     else
            cout<<"\nThe Search Element "<<Sno<<" is available";
 }
```

3.b) An array A[10][20] is stored in the memory with each element requiring 4 bytes of storage. If the base address of the array in the memory is 400, determine the location of A[8][13] when the array VAL is stored (i) Row major (ii) Column major.

Solution: Children, Try this answer.

3.c) Write a user-defined function in C++ to find and display the multiplication of row elements of two dimensional array A[4][6] containing integers.

```
void rowmul( )
{    int A[4][6],i,j,rowmul;
    cout<<"\nEnter any 24 values…";
```

```
    for(i=0;i<4;i++)
        for(j=0;j<6;j++)
            cin>>A[i][j];

   for(i=0;i<4;i++)
    {    rowmul=1;
        for(j=0;j<6;j++)
             rowmul=rowmul*A[i][j];
        cout<<"\nThe multiplication of "<<i+1<<" row = "<<rowmul;
    }
   }
```

DELHI 1998

3.a) Suppose an array P containing float is arranged in ascending order. Write a user defined function in C++ to search for one float from p with the help of binary search method. The function should return an integer 0 to show absence of the number in the array. The function should have the parameters as (1) an array P (2) the number DATA to be searched (3) number of elements N.

```
int BinSearch(float P[ ], float DATA, int N)
{    int  l=0,u=N-1,m;
     while(l<=u)
     {    m=(l+u)/2;
          if (DATA= = P[m])
               return 1;
          else if(DATA<P[m])
               u=m-1;
          else
               l=m+1;
     }
     return 0;
 }
```

3.b) An array T[15][10] is stored in the memory with each element requiring 2 bytes of storage. If the base address of T is 2000, determine the location of T[7][8] when the array VAL is stored (i) Row major (ii) Column major.

Solution: Children, Try this as an assignment.

3.c) Write a user-defined function in C++ to find and display the sum of diagonal elements from a 2D array R[7][7] containing integers.

```
void displaysum( )
{  int i,j,D1=0,D2=0,R[7][7];
   cout<<"\nEnter any 49 values….";
   for(i=0;i<7;i++)
       for(j=0;j<7;j++)
        {  cin>>R[i][j];
           if(i= = j)
               D1=D1+R[i][j];
           else if ((i+j)= =(size-1))
               D2=D2+R[i][j];
        }
    cout<<"\nThe sum of the elements of the Main Diagonal = "<<D1;
    cout<<"\nThe sum of the elements of the Other Diagonal = "<<D2;
}
```

" If WEALTH is lost, nothing is lost.
If HEALTH is lost, something is lost.
If CHARACTER is lost, everything is lost."

"The fear of the Lord is the beginning of wisdom"