

## —2-D ARRAYS USING FUNCTIONS—

### 1. WAP to find sum of odd and even elements in m\*n matrix

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

void input( int [100 ][100] , int, int);
void display(int [ 100][100], int, int);
void sum_odd_even(int [ 100][100], int, int);

int main(){

    int a[100][100], row, column;
    printf("Enter the row of matrix:");
    scanf("%d", &row);
    printf("Enter the column of matrix:");
    scanf("%d", &column);
    input(a, row, column);
    printf("\nThe Array Elements are:\n");
    display(a, row, column);
    sum_odd_even( a, row, column);
    getch();
    return 0;
}
```

```
void input( int a[ 100][100] , int m, int n){

    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<n; j++){
            printf("Enter a[%d][%d] element:",i, j);
            scanf("%d", &a[i][j]);
        }
    }
}
```

---

```

void display( int a[ 100][100] , int m, int n){
    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<n; j++){
            printf("%d\t", a[i][j]);
        }
        printf("\n");
    }
}

void sum_odd_even(int a[ 100][100], int m, int n){
    int i, j, osum=0,esum=0;
    for(i=0;i<m; i++){
        for(j=0;j<n; j++){
            if(a[i][j]%2==0){
                esum=esum+a[i][j];
            }
            else{
                osum=osum+a[i][j];
            }
        }
    }
    printf("\nEven Sum=%d and Odd Sum=%d", esum, osum);
}

```

**2. WAP to find sum of main diagonal and sum of right diagonal of n\*n square matrix.**

```

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

void input( int [100 ][100] , int);
void display(int [ 100][100], int);
void sum_diagonal(int [ 100][100], int);
int main(){

    int a[100][100], row, column;
    printf("Enter the row of matrix:");

```

---

```
scanf("%d", &row);
printf("Enter the column of matrix:");
scanf("%d", &column);

//for square matrix rows should be equal to columns
if(row!=column){
    printf("Matrix should be square.");
    exit(0);
}

input(a, row);
printf("\nThe Array Elements are:\n");
display(a, row);
sum_diagonal ( a, row);
getch();
return 0;
}

void input( int a[ 100][100] , int m){
    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<m; j++){
            printf("Enter a[%d][%d] element:",i, j);
            scanf("%d", &a[i][j]);
        }
    }
}

void display( int a[ 100][100] , int m){
    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<m; j++){
            printf("%d\t", a[i][j]);
        }
        printf("\n");
    }
}
```

```

void sum_diagonal(int a[ 100][100], int m){
    int i, j, main_diagonal=0, right_diagonal=0;

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

            if(i==j){
                main_diagonal = main_diagonal +a[i][j];
            }
            if((i+j)==(m-1)){
                right_diagonal = right_diagonal +a[i][j];
            }
        }
    }
    printf("\nThe sum of main diagonal elements is:%d", main_diagonal);
    printf("\nThe sum of right diagonal elements is:%d", right_diagonal);
}

```

### **3. WAP to find highest and lowest element of m\*n matrix..**

```

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

void input( int [100 ][100] , int, int);
void display(int [ 100][100], int, int);
void max_min(int [ 100][100], int, int );

int main(){

    int a[100][100], row, column, max, min;
    printf("Enter the row of matrix:");
    scanf("%d", &row);
    printf("Enter the column of matrix:");
    scanf("%d", &column);
    input(a, row, column);
    printf("\nThe Array Elements are:\n");
    display(a, row, column);
    max_min( a, row, column);
    getch();
    return 0;
}

```

---

```
void input( int a[ 100][100] , int m, int n){  
    int i, j;  
    for(i=0;i<m; i++){  
        for(j=0;j<n; j++){  
            printf("Enter a[%d][%d] element:",i, j);  
            scanf("%d", &a[i][j]);  
        }  
    }  
}
```

```
void display(int a[ 100][100] , int m, int n){  
    int i, j;  
    for(i=0;i<m; i++){  
        for(j=0;j<n; j++){  
            printf("%d\t", a[i][j]);  
        }  
        printf("\n");  
    }  
}
```

```
void max_min(int a[ 100][100] , int m, int n){  
    int i, j, max=a[0][0], min=a[0][0];  
  
    for(i=0; i<m; i++){  
        for(j=0; j<n; j++){  
            if(a[i][j]>max){  
                max=a[i][j];  
            }  
            if(a[i][j]<min){  
                min=a[i][j];  
            }  
        }  
    }  
    printf("\nMaximum is:%d and Minimum is:%d", max, min);  
}
```

---

**4. WAP to add two n\*n square matrix**

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

void input( int [ 100 ][100] , int);
void display(int [ 100][100], int);
void add(int[100][100], int[100][100],int[100][100] ,int);

int main(){
    int a[100][100], b[100][100],c[100][100], n;
    printf("Enter size of square matrix:");
    scanf("%d", &n);

    printf("\nEnter First Matrix:\n");
    input(a, n);

    printf("\nEnter Second Matrix:\n");
    input(b, n);

    printf("\nFirst matrix:\n");
    display(a, n);

    printf("\nSecond matrix :\n");
    display(b, n);

    add(a, b, c,n);
    printf("\nAdded matrix:\n");
    display(c, n);
    getch();
    return 0;
}

void input( int a[ 100][100] , int n){
    int i, j;
    for(i=0;i<n; i++){
        for(j=0;j<n; j++){
            printf("Enter a[%d][%d] element:",i, j);
            scanf("%d", &a[ i ][ j ]);
        }
    }
}
```

---

```

void display( int a[ 100][100] , int n){
    int i, j;
    for(i=0;i<n; i++){
        for(j=0;j<n; j++){
            printf("%d\t", a[i][j]);
        }
        printf("\n");
    }
}

void add(int a[100][100], int b[100][100],int c[100][100] ,int n){

int i, j;
for(i=0;i<n; i++){
    for(j=0;j<n; j++){
        c[i][j]=a[i][j]+b[i][ j];
    }
}
}

```

### **5. WAP to multiply two m\*n square matrix**

```

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

void input( int [100 ][100] , int, int);
void display(int [ 100][100], int, int);
void mul(int[100][100], int[100][100],int[100][100] ,int, int);
int main(){
    int a[100][100], b[100][100],c[100][100], r1, c1, r2,c2;
    printf("Enter the row of first matrix:");
    scanf("%d", &r1);
    printf("Enter the column of first matrix:");
    scanf("%d", &c1);
    printf("Enter the row of second matrix:");
    scanf("%d", &r2);
    printf("Enter the column of second matrix:");
    scanf("%d", &c2);
    if(c1!=r2){
        printf("Matrix multiplication not possible.");
        exit(0);
    }
}

```

---

```
printf("\nEnter First Matrix:\n");
input(a, r1, c1);

printf("\nEnter Second Matrix:\n");
input(b, r2, c2);

printf("\nFirst array elements are:\n");
display(a, r1, c1);

printf("\nSecond array elements are:\n");
display(b, r2, c2);

mul(a, b, c, r1, c2);
printf("\nMultiplied array elements are:\n");
display(c, r1, c2);

getch();
return 0;
}
```

```
void input( int a[ 100][100] , int m, int n){
    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<n; j++){
            printf("Enter a[%d][%d] element:",i, j);
            scanf("%d", &a[i][j]);
        }
    }
}
```

```
void display( int a[ 100][100] ,int m,  int n){
    int i, j;
    for(i=0;i<m; i++){
        for(j=0;j<n; j++){
            printf("%d\t", a[i][j]);
        }
        printf("\n");
    }
}
```

---

```

void mul(int a[100][100], int b[100][100],int c[100][100] ,int m, int n){

int i, j,k, sum=0;
    for(i=0; i<m; i++){
        for(j=0; j<n; j++){
            for(k=0; k<n; k++){
                sum=sum+a[i][k]*b[k][j];
            }
            c[i][j]=sum;
            sum=0;
        }
    }
}

```

**6. WAP to find transpose of  $m \times n$  square matrix**

```

#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

void input( int [ 100 ][100] , int, int);
void display(int [ 100][100], int, int);
void transpose(int[100][100] ,int[100][100], int, int);
int main(){

    int a[100][100], b[100][100], m, n;
    printf("Enter the row of matrix:");
    scanf("%d", &m);
    printf("Enter the column of matrix:");
    scanf("%d", &n);

    printf("\nEnter Matrix:\n");
    input(a, m, n);

    printf("\nThe matrix is:\n");
    display(a, m, n);

    transpose(a, b, m, n);
    printf("\nTranspose matrix is :\n");
    //for transpose matrix row=n and column=m
    display (b, n, m);
    getch();
    return 0;
}

```

---

```
void input( int a[ 100][100] , int m, int n){  
  
    int i, j;  
    for(i=0;i<m; i++){  
        for(j=0;j<n; j++){  
            printf("Enter a[%d][%d] element:",i, j);  
            scanf("%d", &a[i][j]);  
        }  
    }  
}
```

```
void display( int a[ 100][100] ,int m, int n){  
  
    int i, j;  
    for(i=0;i<m; i++){  
        for(j=0;j<n; j++){  
            printf("%d\t", a[i][j]);  
        }  
        printf("\n");  
    }  
}
```

```
void transpose(int a[100][100], int b[100][100], int m, int n){  
  
    int i, j;  
    for(i=0; i<m; i++){  
        for(j=0; j<n; j++){  
            b[j][i]=a[i][j];  
        }  
    }  
}
```

---