Practical 1 (Array ADT)

CODE :

#include <stdio.h>

#define M 5

#define N 5

struct matrix{

int a[M][N];

int m;

int n;

};

void initialise(struct matrix \*A1,struct matrix \*A2)

{

printf("Enter number of rows and column for first matrix: ");

scanf("%d %d",&A1->m,&A1->n);

printf("Enter number of rows and column for second matrix: ");

scanf("%d %d",&A2->m,&A2->n);

}

int getinput()

{

int x;

scanf("%d",&x);

return x;

}

void create(struct matrix \*A)

{

int i,k,j;

for(i=0;i<A->m;i++)

{

for(j=0;j<A->n;j++)

{

A->a[i][j]=getinput();

}

}

}

void display (struct matrix A)

{

int i,k,j;

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

{

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

{

printf("%d ",A.a[i][j]);

}

printf("\n");

}

}

void row\_major(struct matrix A)

{

int i,j;

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

{

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

{

printf("%d ",A.a[i][j]);

}

}

printf("\n");

}

void col\_major(struct matrix A)

{

for(int i=0;i<A.m;i++)

{

for(int j=0;j<A.n;j++)

{

printf("%d ",A.a[j][i]);

}

}

printf("\n");

}

void addition(struct matrix \*A1,struct matrix\*A2,struct matrix\*A3)

{

int i,j;

A3->m=A1->m;

A3->n=A1->n;

if(A1->m==A2->m && A1->n==A1->m)

{

for(i=0;i<A1->m;i++)

{

for(j=0;j<A1->n;j++)

{

A3->a[i][j]=A1->a[i][j]+A2->a[i][j];

}

}

}

else

{

printf("Cannot be added\n");

}

//display

for(i=0;i<A3->m;i++)

{

for(j=0;j<A3->n;j++)

{

printf("%d ",A3->a[i][j]);

}

printf("\n");

}

}

void multiply(struct matrix \*A1,struct matrix \*A2,struct matrix \*A4)

{

A4->m=A1->m;

A4->n=A1->n;

int i,j,k;

if(A1->n==A2->m)

{

for(i=0;i<A1->m;i++)

{

for(j=0;j<A1->n;j++)

{

A4->a[i][j]=0;

for(k=0;k<A1->n;k++)

{

A4->a[i][j]+=A1->a[i][k]\*A2->a[k][j];

}

}

}

}

else

{

printf("Matrix multiplication not possible\n");

}

//display

for(i=0;i<A4->m;i++)

{

for(j=0;j<A4->n;j++)

{

printf("%d ",A4->a[i][j]);

}

printf("\n");

}

}

int main()

{

struct matrix A1,A2,A3,A4;

initialise(&A1,&A2);

printf("Create first matrix\n");

create(&A1);

printf("Create second matrix\n");

create(&A2);

printf("First matrix is :\n");

display(A1);

printf("second matrix is :\n");

display(A2);

printf("Row major of A1:\n");

row\_major(A1);

printf("col major of A1:\n");

col\_major(A1);

printf("Addition of first and second matrix:\n");

addition(&A1,&A2,&A3);

printf("Multiplication of first and second matrix:\n");

multiply(&A1,&A2,&A4);

    return 0;

}