**#2. Array Operations**

**Roll Number:CB.EN.P2EBS22002**

**Date of Submission:4-11-2022**

**Aim:**

To perform following array operations using C Programming:

1. Create a 3-dimensional matrix of dimension 3x3x3. The entries inside the matrix can be as shown below,
2. Perform matrix multiplication of two 3x3 matrix. Matrix data shall be acquired from the user. User defined function can be used for matrix multiplication operation.

**Tools Required:**

Text editor with C Compiler.

**Experiment:**

1. **create a 3- dimensional matrix of dimension 3x3x3.The entries inside the matrix can be shown as**

Code

#include<stdio.h>

int i,j,k; //variables for nested for loops

int main()

{

int arr[2][3][3]; //array declaration

printf("enter the values in the array: \n");

for(i=1;i<2;i++) //represents block

{

for(j=1;j<=3;j++) //represents rows

{

for(k=1;k<=3;k++) //represents columns

{

printf("the value at arr[%d][%d][%d]: ",i,j,k);

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

}

}

}

printf("printing the values in array: \n");

for(i=1;i<2;i++)

{

for(j=1;j<=3;j++)

{

for(k=1;k<=3;k++)

{

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

if(k==3)

{

printf("\n");

}

}

}

printf("\n");

}

for(i=1;i<2;i++)

{

for(j=1;j<=3;j++)

{

for(k=1;k<=3;k++)

{

printf("%d ",2\*arr[i][j][k]);

if(k==3)

{

printf("\n");

}

}

}

printf("\n");

}

for(i=1;i<2;i++)

{

for(j=1;j<=3;j++)

{

for(k=1;k<=3;k++)

{

printf("%d ",2\*2\*arr[i][j][k]);

if(k==3)

{

printf("\n");

}

}

}

printf("\n");

}

return 0;

}

i)Result

enter the values in the array:

the value at arr[1][1][1]: 0

the value at arr[1][1][2]: 1

the value at arr[1][1][3]: 2

the value at arr[1][2][1]: 3

the value at arr[1][2][2]: 4

the value at arr[1][2][3]: 5

the value at arr[1][3][1]: 6

the value at arr[1][3][2]: 7

the value at arr[1][3][3]: 8

printing the values in array:

0 1 2

3 4 5

6 7 8

0 2 4

6 8 10

12 14 16

0 4 8

12 16 20

24 28 32

ii)Code

#include<stdio.h>

void multiply(int r1, int c1, int r2, int c2,float a[][c1],float b[][c2])

{

int i,j,k;

float mul[r1][c2];

for(i=0;i< r1;i++)

{

for(j=0;j< c2;j++)

{

mul[i][j] = 0;

for(k=0;k< r2;k++)

{

mul[i][j] = mul[i][j] + a[i][k]\*b[k][j];

}

}

}

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

for(i=0;i< r1;i++)

{

for(j=0;j< c2;j++)

{

printf("%f\t", mul[i][j]);

}

printf("\n");

}

}

int main()

{

int i,j,k,r1,c1,r2,c2;

printf("Enter row and column of first matrix\n");

scanf("%d%d", &r1, &c1);

printf("Enter row and column of second matrix\n");

scanf("%d%d", &r2, &c2);

float a[r1][c1], b[r2][c2];

if(c1==r2)

{

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

for(i=0;i< r1;i++)

{

for(j=0;j< c1;j++)

{

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

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

}

}

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

for(i=0;i< r2;i++)

{

for(j=0;j< c2;j++)

{

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

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

}

}

multiply(r1,c1,r2,c2,a,b);

}

else

{

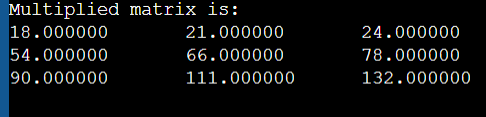
printf("Dimension do not match for multiplication.");

}

return 0;

}

ii)Result



**Inference and Result:**

**Multiplication of matrices and multidimensional array operations were performed and results are computed.**