

Lab 4-2 array 2d

ตอนที่ 1 - จงปรับปรุงโปรแกรม เพื่อให้สามารถรับค่าข้อมูลจำนวนเต็มไปเก็บไว้ในอาร์เรย์ scores โดยใช้คำสั่ง scanf แทนการกำหนดค่า เริ่มต้น โดยให้ผลของโปรแกรมเหมือนเดิม - เพิ่มการคำนวณ หาค่าเฉลี่ยรวม ของทุกจำนวนในอาร์เรย์ และแสดงผลลัพธ์ ก่อนจบโปรแกรม

```
#include<stdio.h>
int main()
{
    int i, j, scores[3][5];
    float sum, avg;

    for (i=0; i<3; i++)
    {
        for (j=0, sum=0.0; j<5; j++) {
            scanf("%d", &scores[i][j]);
        }
    }

    for (i=0; i<3; i++)
    {
        for (j=0, sum=0.0, avg=0.0; j<5; j++) {
            sum = sum+scores[i][j];
        }
        avg = sum/5;
        printf("Sum of row %d = %.2f \n", i+1, sum);
        printf("Average of row %d = %.2f \n\n", i+1, avg);
    }

    return 0;
}
```

```
54 54 87 6 25 39 15 28 55 99 33 24 15 24 25
Sum of row 1 = 226.00
Average of row 1 = 45.20

Sum of row 2 = 236.00
Average of row 2 = 47.20

Sum of row 3 = 121.00
Average of row 3 = 24.20

-----
Process exited after 50.52 seconds with return value 0
Press any key to continue . . .
```

ตอนที่ 2 checkpoint 2

- จงแก้ไข เพิ่มเติมโปรแกรม เพื่อให้สามารถ ทำการคูณแต่ละสมาชิกของเมตริกส์ A และ B ในตำแหน่งที่ตรงกันได้ เช่น

$A = \begin{bmatrix} 2 & 3 & 5 \end{bmatrix}$, $B = \begin{bmatrix} 4 & 0 & 8 \end{bmatrix}$, $C = A * B = \begin{bmatrix} 8 & 0 & 40 \end{bmatrix}$ หมายถึง A,B และ C มีขนาดเท่ากัน

```
#include<stdio.h>
#define MAX 10
int main()
{
    int matA[MAX][MAX] = {}, matB[MAX][MAX] = {}, matC[MAX][MAX] = {};
    int ra, ca, rb, cb, i, j;
    printf("Matrix A\n");
    printf("Enter row : ", ra);
    scanf("%d", &ra);
    printf("Enter column : ", ca);
    scanf("%d", &ca);
    for(i=0; i<ra; i++){
        for(j=0; j<ca; j++){
            printf("matA(%d,%d) = ", i+1, j+1);
            scanf("%d", &matA[i][j]);
        }
    }

    for(i=0; i<ra; i++){
        for(j=0; j<ca; j++){
            printf("%5d", matA[i][j]);
        }
        printf("\n");
    }

    printf("\nMatrix B\n");
    printf("Number of rows = "); scanf("%d", &rb);
    printf("Number of columns = "); scanf("%d", &cb);
    //Input elements of Matrix B (rb x cb)
    for (i=0; i<rb; i++)
    {
        for (j=0; j<cb; j++)
        {
            printf("matB(%d,%d) = ", i+1, j+1);
```

```
Matrix A
Enter row :1
Enter column :3
matA(1,1)= 2
matA(1,2)= 1
matA(1,3)= 3
2 1 3

Matrix B
Number of rows = 1
Number of columns = 3
matB(1,1)=5
matB(1,2)=6
matB(1,3)=-7
5 6 -7
Matrix C = A * B
10 6 -21

-----
Process exited after 15.66 seconds with return value 0
Press any key to continue . . .
```

Checkpoint 3

- จงแก้ไขเพิ่มเติมโปรแกรม เพื่อให้สามารถ ทำการคูณเมตริกส์ ได้ เมตริกส์ $C(m \times p) = A(m \times n) \times B(n \times p)$ เมตริกส์ ที่คูณกันได้ ตัวตั้ง (A) ต้องมีจำนวนหลัก เท่ากับจำนวนแถว ของตัวคูณ (B) และ เมตริกส์ผลคูณ (C) จะมีจำนวนแถว เท่ากับตัวตั้งและจำนวนหลักเท่ากับตัวคูณ

```
#include <stdio.h>
#define MAX 10
int main()
{
    int matA[MAX][MAX]={}, matB[MAX][MAX]={}, matC[MAX][MAX]={};
    int ra,rb,ca,cb,i,j,k,sum;
    printf("Matrix A\n");
    printf("Number of rows = "); scanf("%d",&ra);
    printf("Number of columns = "); scanf("%d",&ca);
    // Input elements of Matrix A(ra x ca)
    for (i=0; i<ra; i++)
    {
        for(j=0; j<ca; j++)
        {
            printf("matA(%d,%d)=",i+1,j+1);
            scanf("%d",&matA[i][j]);
        }
    }
    // Show elements of Matrix A(ra x ca)
    for (i=0; i<ra; i++)
    {
        for(j=0; j<ca; j++) printf("%5d",matA[i][j]);
        printf("\n");
    }
    printf("\nMatrix B\n");
    printf("Number of rows = "); scanf("%d",&rb);
    printf("Number of columns = "); scanf("%d",&cb);
    // Input elements of Matrix B(rb x cb)
    for (i=0; i<rb; i++)
    {
        for(j=0; j<cb; j++)
        {
            printf("matB(%d,%d)=",i+1,j+1);
            scanf("%d",&matB[i][j]);
        }
    }
    // Show elements of Matrix B(rb x cb)
    for (i=0; i<rb; i++)
    {
        for(j=0; j<cb; j++) printf("%5d",matB[i][j]);
        printf("\n");
    }
    // Matrix C = A * B
    for (i=0; i<ra; i++)
    {
        for(j=0; j<cb; j++)
        {
            sum=0;
            for(k=0; k<ca; k++)
            {
                sum+=matA[i][k]*matB[k][j];
            }
            matC[i][j]=sum;
        }
    }
    printf("\nMatrix C = A * B\n");
    for (i=0; i<ra; i++)
    {
        for(j=0; j<cb; j++) printf("%5d",matC[i][j]);
        printf("\n");
    }
}
```

Output:

```
Number of columns = 2
matA(1,1)=2
matA(1,2)=3
matA(2,1)=1
matA(2,2)=2
matA(3,1)=3
matA(3,2)=8
2 3
1 2
3 8

Matrix B
Number of rows = 2
Number of columns = 3
matB(1,1)=1
matB(1,2)=5
matB(1,3)=6
matB(2,1)=-9
matB(2,2)=3
matB(2,3)=5
1 5 6
-9 3 5

Matrix C = A * B
-25 19 27
-17 11 16
-69 39 58
```

การบ้าน

3. มีชั้นวางหนังสืออยู่หลังหนึ่ง มีช้ ONท์ Oงหมด 2 ช้ ON วางหนังสือ ได้ช้ ONละ 5 เล่ม มีการเก็บข้อมูลของหนังสือราคา จำนวนหน้า และ น้ำหนัก ให้ประกาศอาร์เรย์สามมิติ ในการเก็บข้อมูล ดังนี้ `O float books[2][5][3];` - โดยดัชนีตัวแรกอ้างถึง ตำแหน่งของชั้นวางหนังสือ (ชั้นวางที่ 0 หรือ 1) - ดัชนีตัวที่สองคือตำแหน่งหนังสือ (เล่มที่ 0 ถึง 4) - และดัชนีตัวสุดท้ายบ่งชี้ ข้อมูลของหนังสือ (ราคา จำนวนหน้า และ น้ำหนัก) - จงเขียนโปรแกรม เพื่อเก็บและแสดงผลข้อมูล ดังต่อไปนี้

ตำแหน่งชั้น (level)	ตำแหน่งหนังสือ (book position)	ราคา (price)	จำนวนหน้า (pages)	น้ำหนัก (wieght)
ชั้น 0	0	280.50	350	152.5
	1	410.00	280	202.5
	2	1590.75	406	858.7
	3	296.50	310	102.5
	4	1210.50	450	952.0
ชั้น 1	0	80.50	75	45
	1	2015	500	575
	2	120	25	30.75
	3	456.50	150	200
	4	110.50	50	130

```
#include<stdio.h>
int main()
{
    float books[2][5][3] =
    {
        {
            {280.50, 350, 152.5},
            {410.00, 280, 202.5},
            {1590.75, 406, 858.7},
            {296.50, 310, 102.5},
            {1210.50, 450, 952.0},
        },
        {
            {80.50, 75, 45},
            {2015, 500, 575},
            {120.0, 25, 30.75},
            {456.50, 150, 200},
            {110.50, 50, 130},
        }
    };

    int i,j,k;
    float maxPrice = 0, minPage = books[0][0][1], avgWeight=0;
    float sum=0;

    printf("%32s %10s %11s\n", "Price", "Pages", "Weight");
    for(i=0; i<2; i++)
    {
        printf("Level %d\n", i+1);
```

C:\Users\Nitro\Downloads\HW-3.exe

```
Level 1
    Price      Pages      Weight
Book position 1 : 280.50      350      152.50
Book position 2 : 410.00      280      202.50
Book position 3 : 1590.75     406      858.70
Book position 4 : 296.50      310      102.50
Book position 5 : 1210.50     450      952.00
Average Weight of level 0 = 453.64
```

```
Level 2
    Price      Pages      Weight
Book position 1 : 80.50       75       45.00
Book position 2 : 2015.00     500      575.00
Book position 3 : 120.00      25       30.75
Book position 4 : 456.50     150      200.00
Book position 5 : 110.50      50       130.00
Average Weight of level 1 = 196.15
```

```
Max price = 2015.00
Min page = 25
```

```
-----
Process exited after 0.02083 seconds with return value 0
Press any key to continue . . .
```