

BLM1011

BİLGİSAYAR BİLİMLERİNE GİRİŞ

GR.2

2023-2024 GÜZ YARIYILI

DR.ÖĞR.ÜYESİ GÖKSEL BİRİCİK

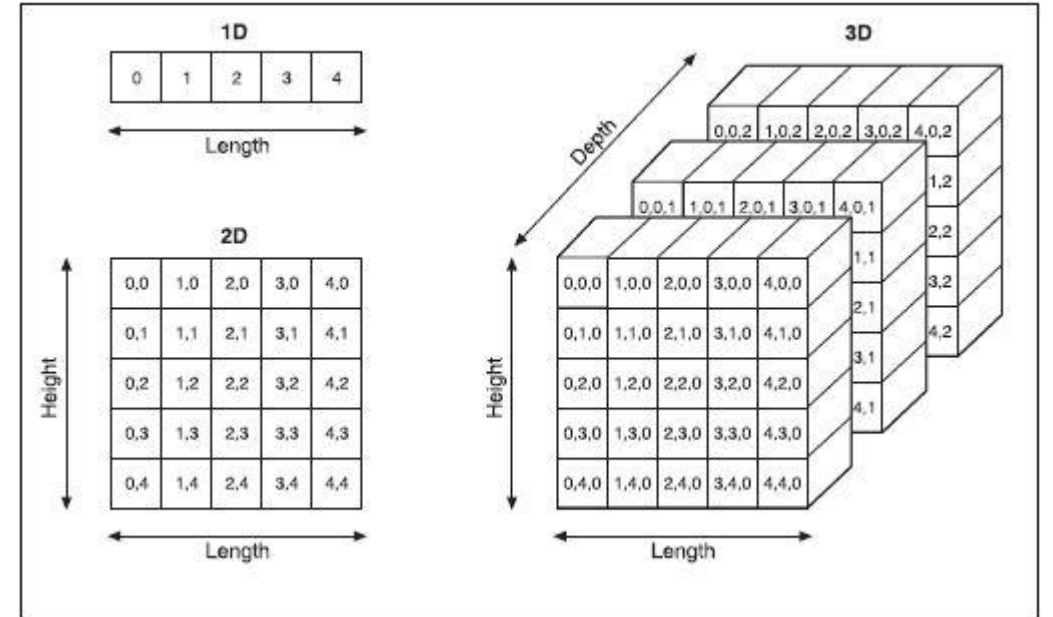
Çok Boyutlu Diziler

Çok Boyutlu Diziler

Aynı türdeki verileri tek bir sıra halinde değil, mantıksal olarak satır-sütun formatında tutulduğu yapılar.

Daha fazla (3 boyutlu, 4 boyutlu, ..., n boyutlu olabilirler.

- 3 boyuttan sonrasını görselleştiremiyoruz.
- Bellekte tek boyut var?
- C'de diziler içsel olarak tek boyutlu dizi biçiminde saklanır.



Çok Boyutlu Dizi Tanımlama

<tür> [boyut1] [boyut2] [boyut3] ...

- Boyutlar sabit ifadeler olmak zorundadır.

int A[5][8];

İfadesi iki boyutlu bir dizidir. Matris olarak da isimlendirilir.

İlk boyut satırları, ikinci boyut sütunları ifade eder.

A[0][0]	A[0][1]	A[0][2]	A[0][3]	A[0][4]	A[0][5]	A[0][6]	A[0][7]
A[1][0]	A[1][1]	A[1][2]	A[1][3]	A[1][4]	A[1][5]	A[1][6]	A[1][7]
A[2][0]	A[2][1]	A[2][2]	A[2][3]	A[2][4]	A[2][5]	A[2][6]	A[2][7]
A[3][0]	A[3][1]	A[3][2]	A[3][3]	A[3][4]	A[3][5]	A[3][6]	A[3][7]
A[4][0]	A[4][1]	A[4][2]	A[4][3]	A[4][4]	A[4][5]	A[4][6]	A[4][7]

Çok Boyutlu Dizi Tanımlama

```
int a[5][2] = { {10, 20},  
                {30, 40},  
                {50, 60},  
                {70, 80},  
                {90, 100}  
            };
```

ya da;

```
int a[5][2] = {10, 20, 30, 40, 50, 60, 70, 80, 90, 100};  
              a00 a01 a10 a11 a20 a21 a30 a31 a40 a41
```

Çok Boyutlu Dizilerin Belleğe Yerleşimi

Düşük boyutu düşük anlamlı adreste bulunacak biçimde sürekli olarak yerleşirler.

```
int a[5][2];
```

...	...
a[0][0]	0.satır
a[0][1]	
a[1][0]	1.satır
a[1][1]	
a[2][0]	2.satır
a[2][1]	
a[3][0]	3.satır
a[3][1]	
a[4][0]	4.satır
a[4][1]	
...	...

Çok Boyutlu Dizilerin Belleğe Yerleşimi

İki boyutlu dizilerin satır elemanları sol taraf değeri değildir. Satırların bellekteki başlangıç adresini gösterirler.

```
int a[5][2];
```

1FC0 adresinden başlayarak yerleşmiş olsun:

a[0], a[1], a[2], a[3], a[4] ifadeleri satırların başlangıç adreslerini gösterir.

Özetle a değişkeni, her elemanı 2 uzunlukta tamsayı dizisi olan 5 uzunluklu bir tamsayı dizisidir.

...	...	
1FC0	a[0][0]	← a[0]
1FC2	a[0][1]	
1FC4	a[1][0]	← a[1]
1FC6	a[1][1]	
1FC8	a[2][0]	← a[2]
1FCA	a[2][1]	
1FCC	a[3][0]	← a[3]
1FCE	a[3][1]	
1FD0	a[4][0]	← a[4]
1FD2	a[4][1]	
...	...	

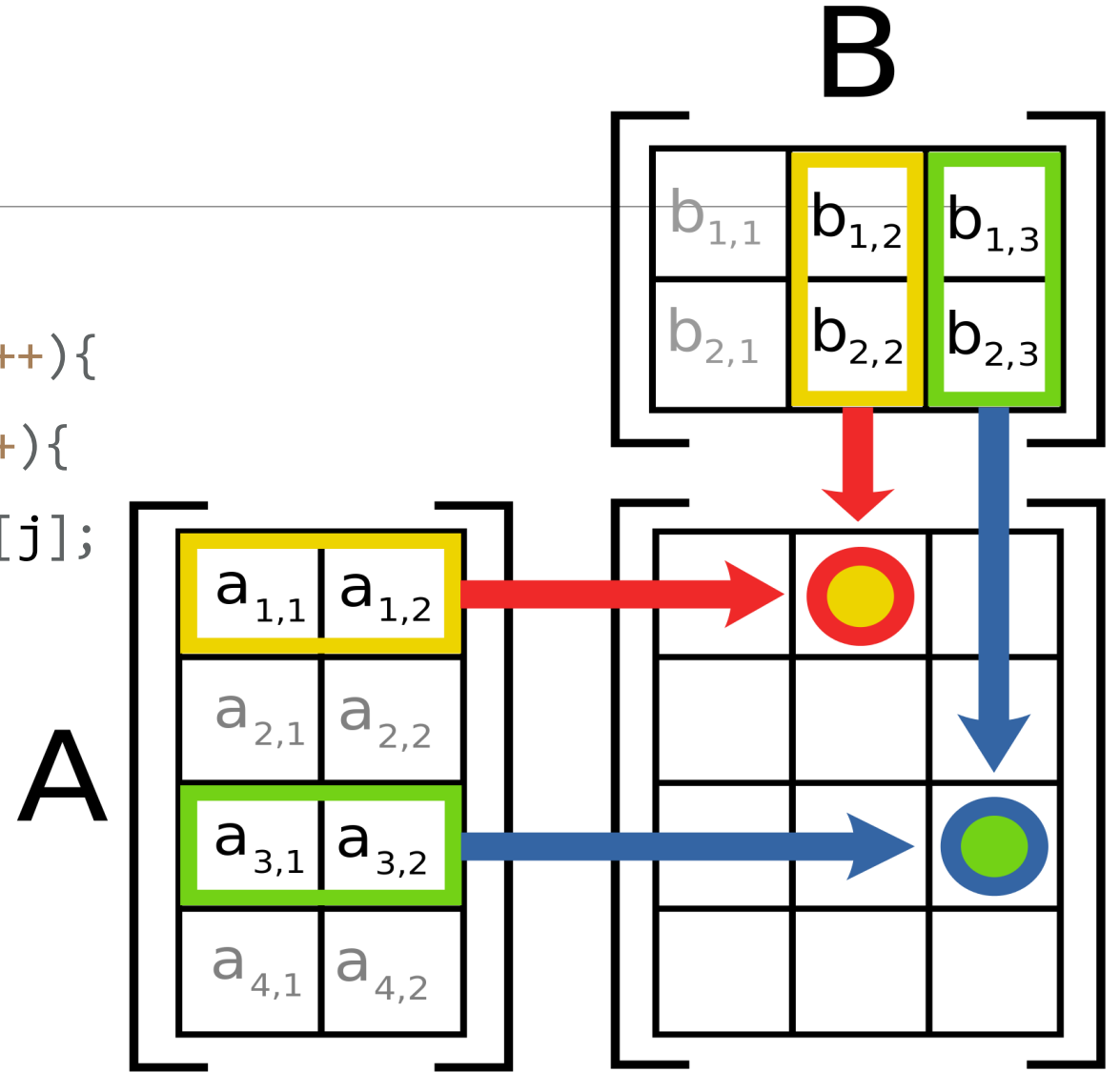
İki Matrisin Elemanlarını Toplama

```
#include <stdio.h>
#define R 5
#define C 2
int main(){
    int a[R][C] = {10, 20, 30, 40, 50, 60, 70, 80, 90, 100};
    int b[R][C] = {5, 10, 15, 20, 25, 30, 35, 40, 45, 50};
    int c[R][C];
    int i,j;
    for (i=0; i<R; i++){
        for(j=0; j<C; j++){
            c[i][j] = a[i][j] + b[i][j];
            printf("[%d][%d] = %d \n",i,j,c[i][j]);
        }
    }
    for (i=0; i<R; i++){
        for(j=0; j<C; j++){
            printf("%4d",c[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

```
[0][0] = 15
[0][1] = 30
[1][0] = 45
[1][1] = 60
[2][0] = 75
[2][1] = 90
[3][0] = 105
[3][1] = 120
[4][0] = 135
[4][1] = 150
15 30
45 60
75 90
105 120
135 150
```


İki Matrisin Çarpımı

```
for(int i = 0; i < M1.length; i++){  
    for(int j = 0; j < M2[0].length; j++){  
        for(int k = 0; k < M2.length; k++){  
            mult[i][j] += M1[i][k] * M2[k][j];  
        }  
    }  
}
```



İki Matrisin Çarpımı

```
#include <stdio.h>

int main()
{
    int m, n, p, q, c, d, k, sum = 0;
    int first[10][10], second[10][10], multiply[10][10];

    printf("İlk matrisin satir sutun sayisini girin:\n");
    scanf("%d%d", &m, &n);
    printf("İlk matris elemanlarini girin:\n");

    for ( c = 0 ; c < m ; c++ )
        for ( d = 0 ; d < n ; d++ )
            scanf("%d", &first[c][d]);

    printf("İkinci matrisin satir sutun sayisini girin:\n");
    scanf("%d%d", &p, &q);

    if ( n != p )
        printf("Boyutlar tutarsız, iki matris carpilamaz.\n");
    else
    {
        printf("İkinci matris elemanlarini girin:\n");

        for ( c = 0 ; c < p ; c++ )
            for ( d = 0 ; d < q ; d++ )
                scanf("%d", &second[c][d]);
```

```
        for ( c = 0 ; c < m ; c++ )
        {
            for ( d = 0 ; d < q ; d++ )
            {
                for ( k = 0 ; k < p ; k++ )
                {
                    sum = sum + first[c][k]*second[k][d];
                }

                multiply[c][d] = sum;
                sum = 0;
            }
        }

        printf("Carpim sonucu:-\n");

        for ( c = 0 ; c < m ; c++ )
        {
            for ( d = 0 ; d < q ; d++ )
                printf("%d\t", multiply[c][d]);

            printf("\n");
        }

        return 0;
    }
}
```

Çok Boyutlu Karakter Dizileri

```
#include <stdio.h>

#define N 5

int main()
{
    char a[N][N]={"ali","veli","ayse","abuzittin"};

    int i;

    for (i=0;i<N;i++){
        puts(a[i]);

    }

    return 0;
}
```

main.c:6:37: warning: initializer-string for array of chars is too long

```
6 |   char a[N][N]={"ali","veli","ayse","abuzittin"};
   |                                     ^~~~~~
```

main.c:6:37: note: (near initialization for ‘a[3]’)

ali

veli

ayse

abuzi

Çok Boyutlu Karakter Dizileri

```
#include <stdio.h>
#define N 10
int main()
{
    char a[N][N]={{'a','l','i','\0'},
                  {'v','e','l','i','\0'},
                  {'a','y','s','e','\0'},
                  {'f','a','t','m','a','\0'}
    };

    int i;

    for (i=0;i<N;i++){
        printf("%d: %s\n",i,a[i]);
    }
    return 0;
}
```

0: ali
1: veli
2: ayse
3: fatma
4:
5:
6:
7:
8:
9:

Çok Boyutlu Karakter Dizileri

```
#include <stdio.h>

int main()
{
    char arr[3][15] = { "yildiz", "teknik", "univertisesi"};
    int i,j,k,kk;
    char temp;

    printf("Once:\n");
    for (int i = 0; i < 3; i++)
        printf("%s\n", arr[i]);

    for (int i = 0; i < 3; i++){
        for(k=0;arr[i][k];k++); //karakter dizi boyutu
        kk=k-1;
        for (int j = 0; j < k/2; j++){
            temp = arr[i][j];
            arr[i][j] = arr[i][kk];
            arr[i][kk] = temp;
            kk--;
        }
    }

    printf("\nSonra:\n");
    for (i = 0; i < 3; i++)
        printf("%s\n", arr[i]);

    return 0;
}
```

Once:

yildiz

teknik

univertisesi

Sonra:

zidliy

kinket

isesitrevinu