

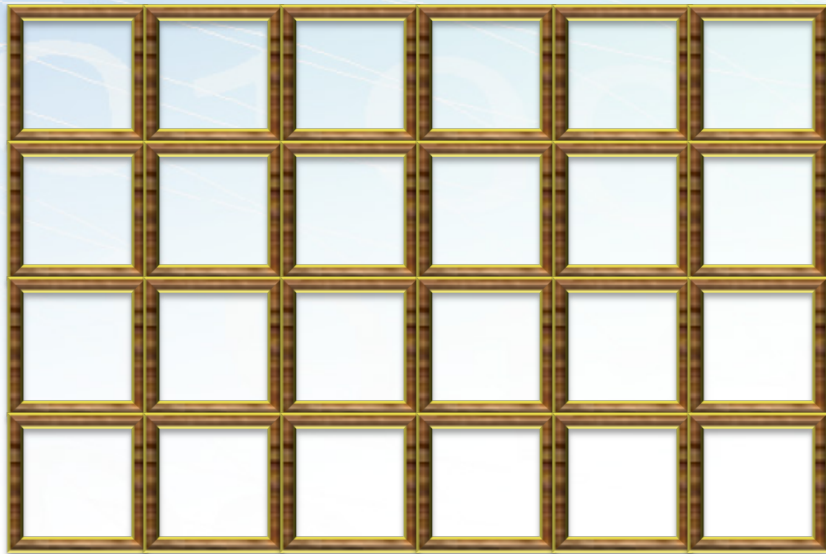
Matrici in C

Matrici

Le **matrici** sono **array** bidimensionali



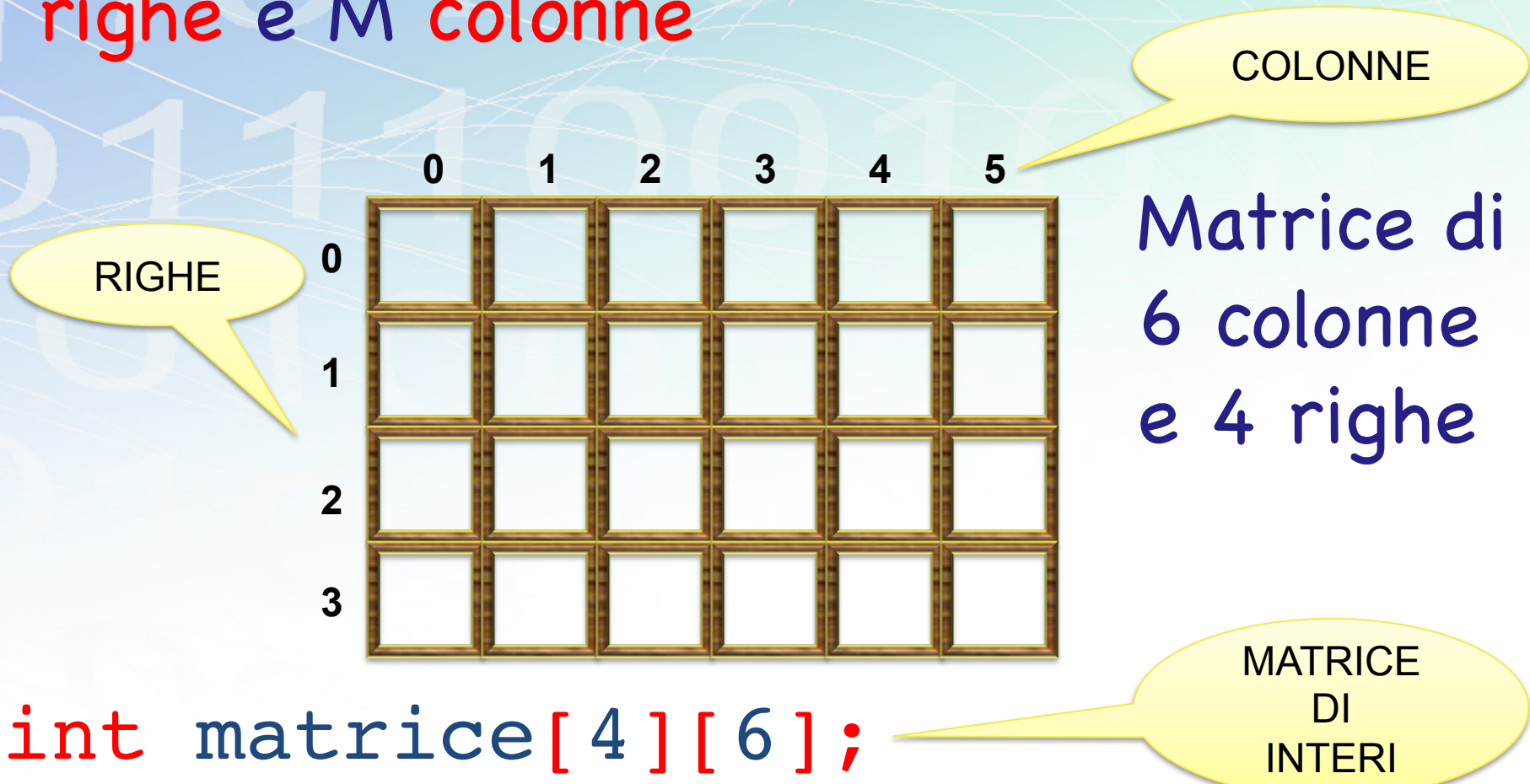
ARRAY



MATRICE

Caratteristiche Matrice

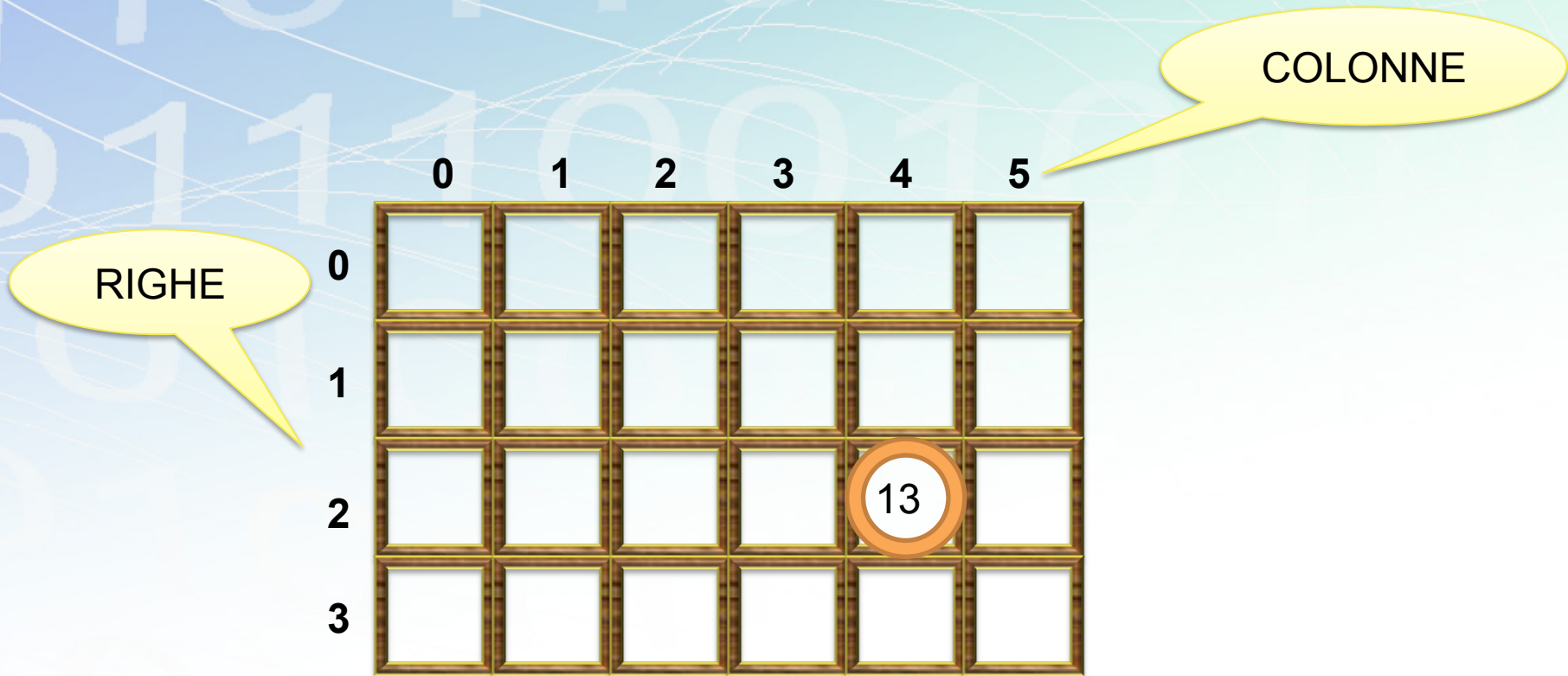
Una matrice è caratterizzata da N
righe e M **colonne**



```
int matrice[4][6];
```

Caratteristiche Matrice

Un elemento è definito da [riga] [colonna]



The diagram illustrates a 4x6 matrix. The rows are indexed 0 to 3 on the left, and the columns are indexed 0 to 5 on top. A yellow speech bubble labeled 'RIGHE' points to the row indices, and another yellow speech bubble labeled 'COLONNE' points to the column indices. The element at row 2, column 4 is highlighted with an orange circle and contains the value 13.

	0	1	2	3	4	5
0						
1						
2					13	
3						

```
matrice[2][4] = 13;
```


Usare una matrice

Per manipolare tutti gli elementi di una matrice in genere si usano due cicli for annidati:

```
for(i=0; i<N; i++) /* righe */
{
    for(j=0; j<M; j++) /* colonne */
    {
        printf("%d", matrice[i][j]);
    }
}
```

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

```
int main(){
```

```
int i,j,matrice[10][10];
```

```
//Cicli per riempire la matrice
```

```
for(i=0;i<10;i++){
```

```
    for(j=0;j<10;j++){
```

```
        matrice[i][j] = i*j;
```

```
    }
```

```
}
```

```
//Cicli per leggere la matrice
```

```
for(i=0;i<10;i++){
```

```
    for(j=0;j<10;j++){
```

```
        printf(" %2d ",matrice[i][j]);
```

```
    }
```

```
    printf("\n");
```

```
}
```

```
return 0;
```

```
}
```

Esercitazione

- Utilizzando l'esempio di prima, modificare il codice per avere un programmino in C che chieda in input due numeri, le righe e le colonne che comporranno una matrice.
- Memorizzare un carattere random (`srand()%255`) in ogni posizione della matrice e poi visualizzarla