Multidimensional Array

"Array of Array"

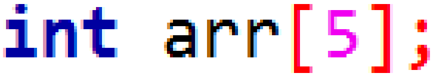
Prerequisite: Array

Find more contents at
https://sites.google.com/view/cse105june18/home

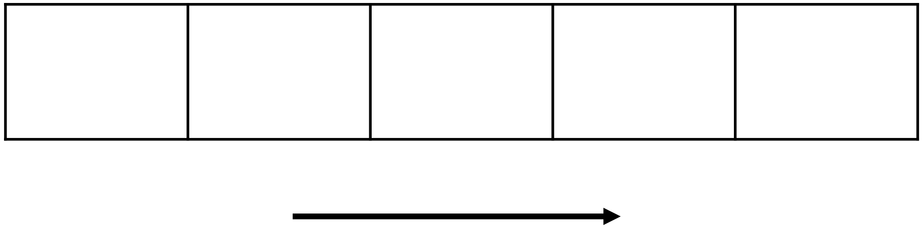
Md. Saidul Hoque Anik
onix.hoque.mist@gmail.com

Prelude

Until now, we've used 1-dimensional array

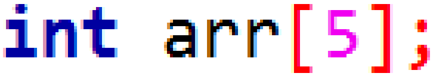


```
int arr[5];
```



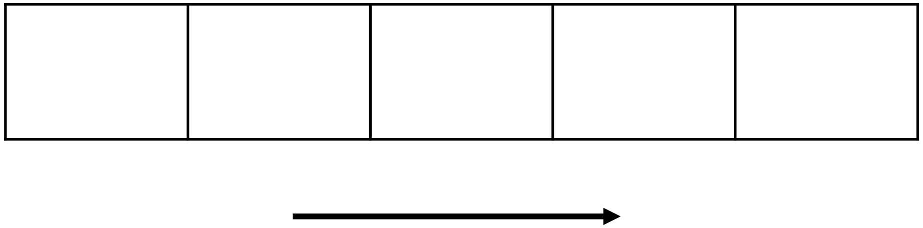
Prelude

Until now, we’ve used 1-dimensional array

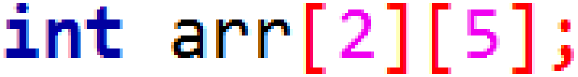


```
int arr[5];
```

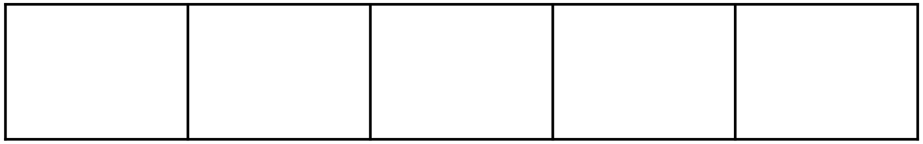
Number of columns



Prelude

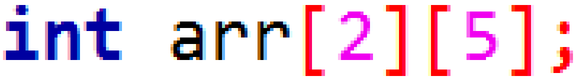


```
int arr[2][5];
```

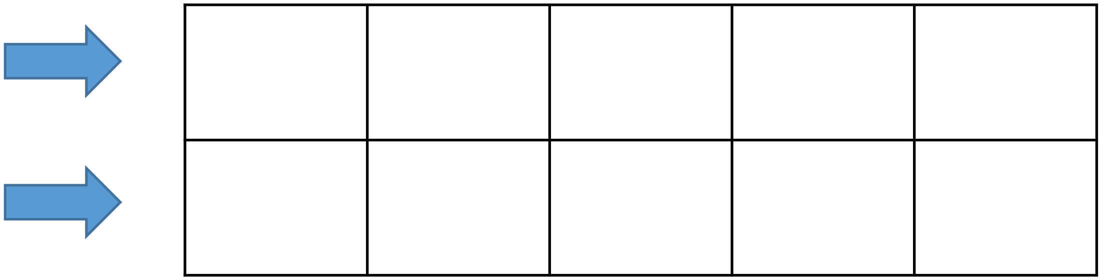


Prelude

A 2-d array

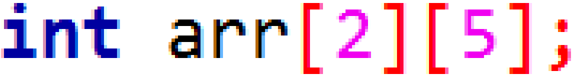


```
int arr[2][5];
```

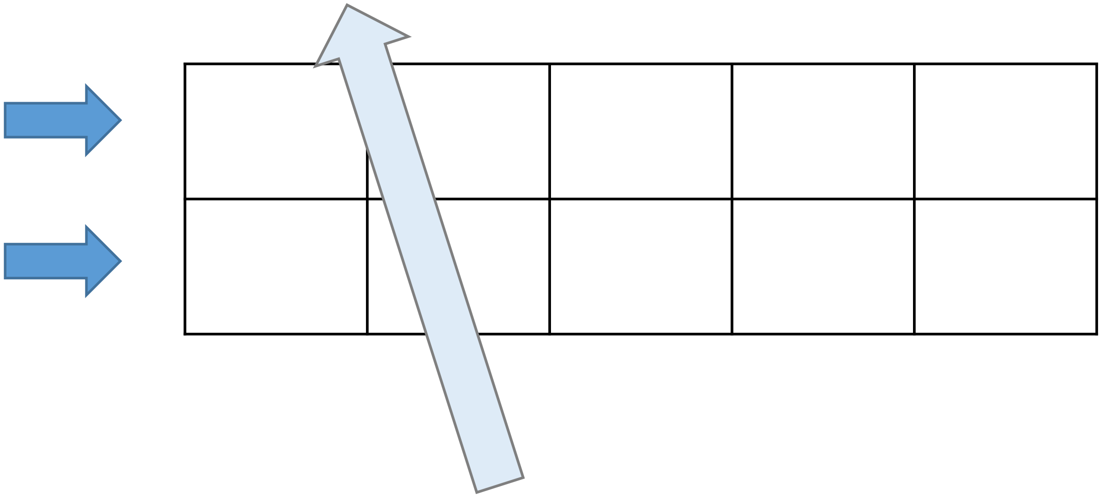


Prelude

A 2-d array



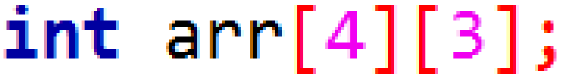
```
int arr[2][5];
```



Number of rows

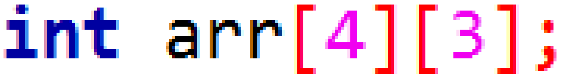
Prelude

How will it look like?

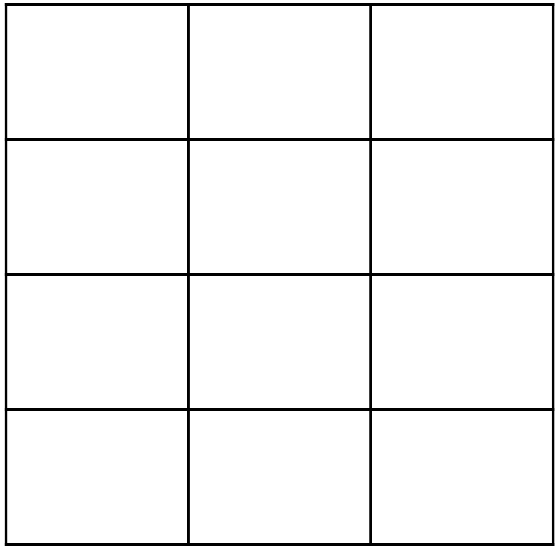


```
int arr[4][3];
```

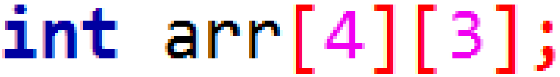
Prelude



```
int arr[4][3];
```



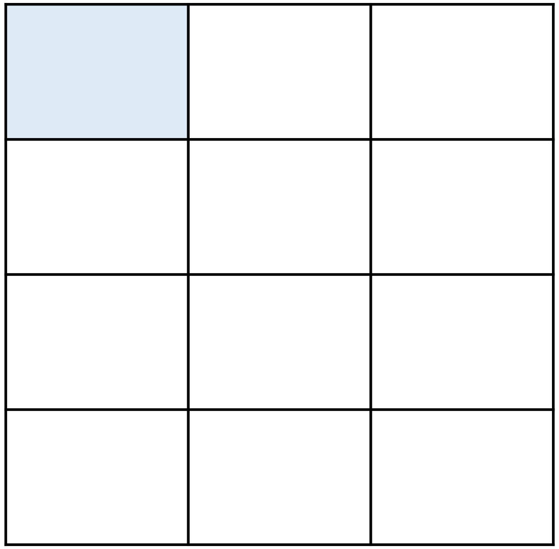
2-D Array



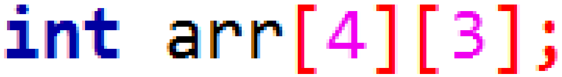
```
int arr[4][3];
```

How can we access this element?

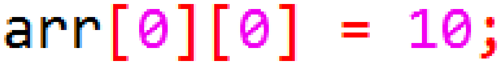
- Row?
- Column?



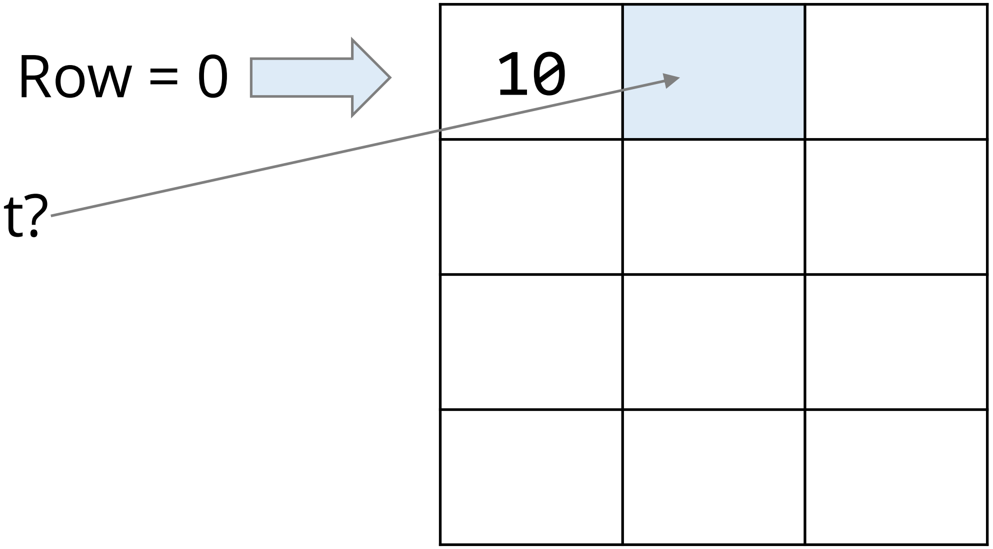
2-D Array



```
int arr[4][3];
```



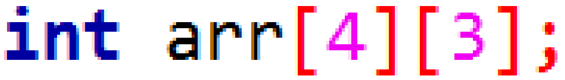
```
arr[0][0] = 10;
```



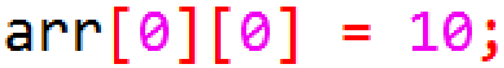
How about this element?

- Row?
- Column?

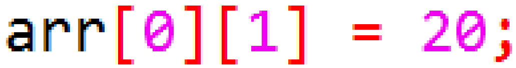
2-D Array



```
int arr[4][3];
```



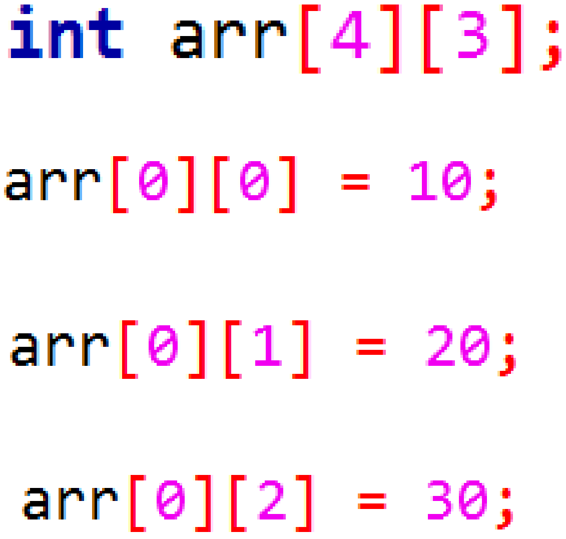
```
arr[0][0] = 10;
```



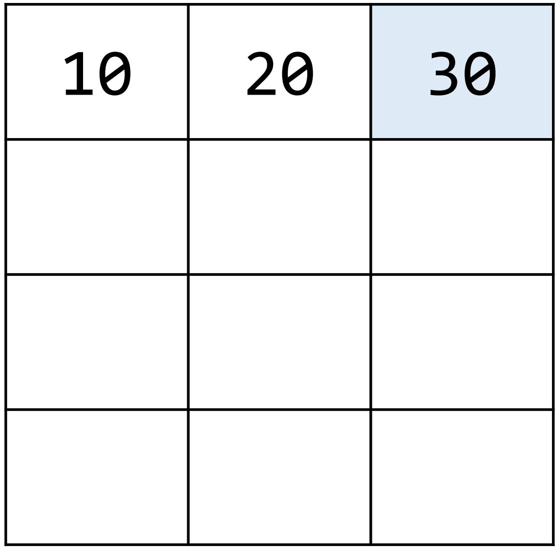
```
arr[0][1] = 20;
```

10	20	

2-D Array

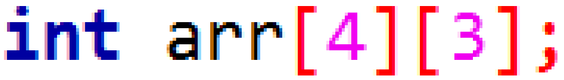


```
int arr[4][3];

arr[0][0] = 10;

arr[0][1] = 20;

arr[0][2] = 30;
```

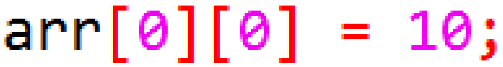


10	20	30

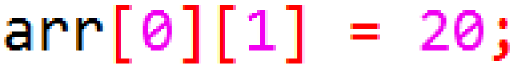
2-D Array



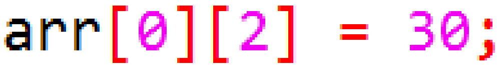
```
int arr[4][3];
```



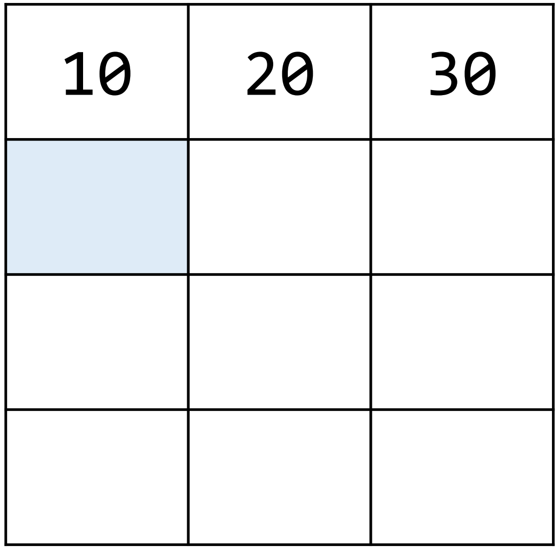
```
arr[0][0] = 10;
```



```
arr[0][1] = 20;
```



```
arr[0][2] = 30;
```



10	20	30

2-D Array

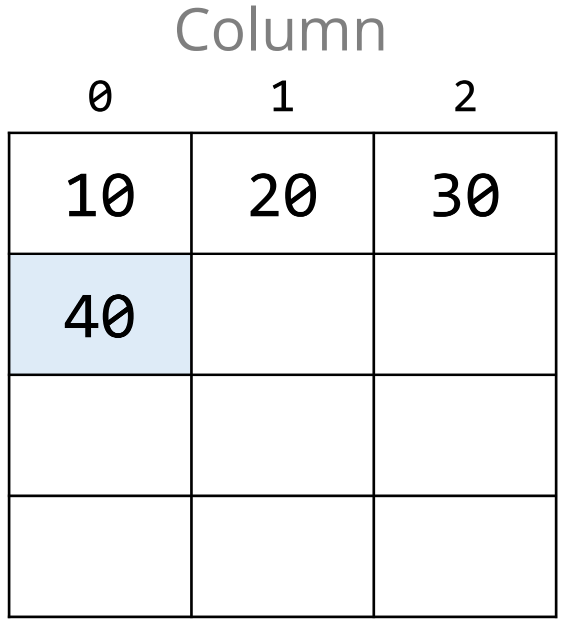
```
int arr[4][3];

arr[0][0] = 10;

arr[0][1] = 20;

arr[0][2] = 30;

arr[1][0] = 40;
```

10	20	30
40		

2-D Array

```
int arr[4][3];

arr[0][0] = 10;

arr[0][1] = 20;

arr[0][2] = 30;

arr[1][0] = 40;
```

Column

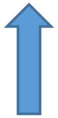


Column

0	1	2
10	20	30
40		

2-D Array

```
int arr[4][3];

arr[0][0] = 10;

arr[0][1] = 20;

arr[0][2] = 30;

arr[1][0] = 40;
```

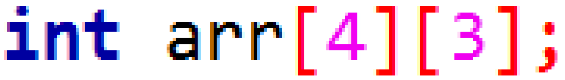


Row

Column

Row	0	1	2
0	10	20	30
1	40		
2			
3			

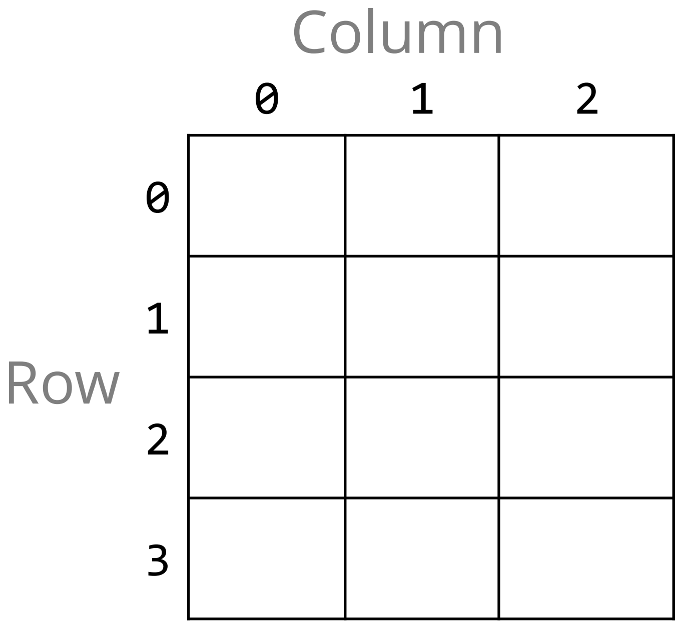
2-D Array



```
int arr[4][3];
```

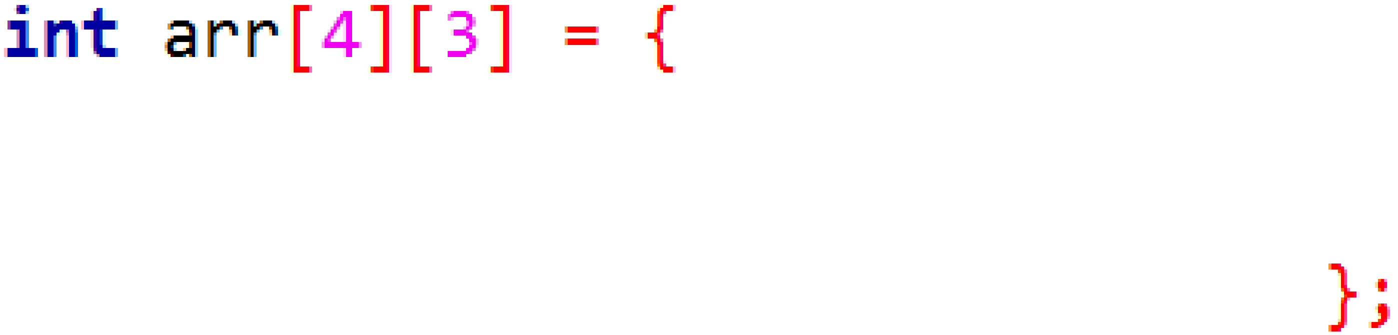
Task(a): Use `for loop` to take input from user and populate the 2-d array.

Task(b): print the 2-d matrix into console.



2-D Array

Initialization



```
int arr[4][3] = {

                                                   };
```

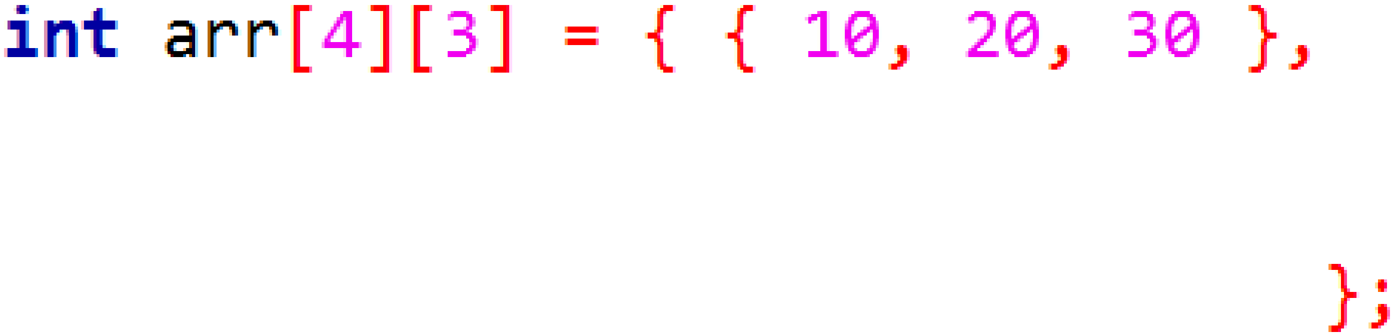
Column

Row

	0	1	2
0			
1			
2			
3			

2-D Array

Initialization



```
int arr[4][3] = { { 10, 20, 30 },

                                                  };
```

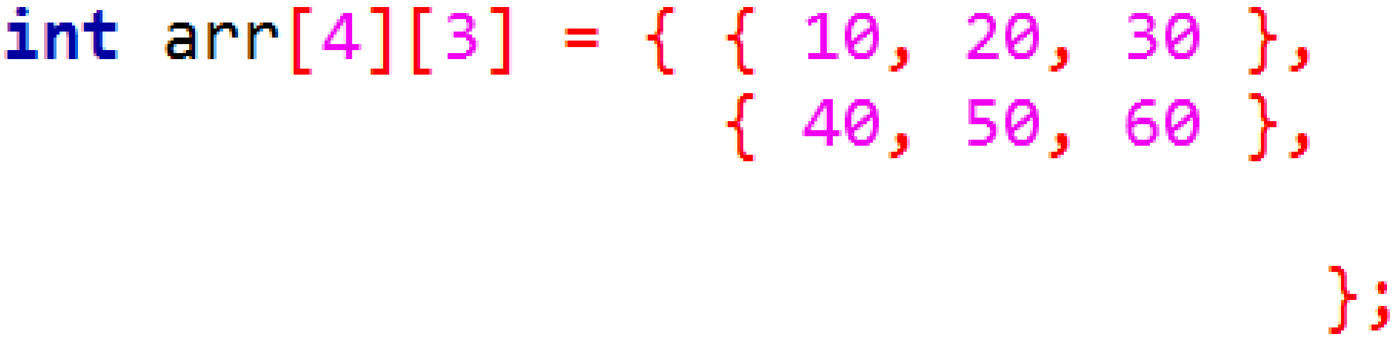
Column

Row

	0	1	2
0	10	20	30
1			
2			
3			

2-D Array

Initialization



```
int arr[4][3] = { { 10, 20, 30 },
                  { 40, 50, 60 },

                                 };
```

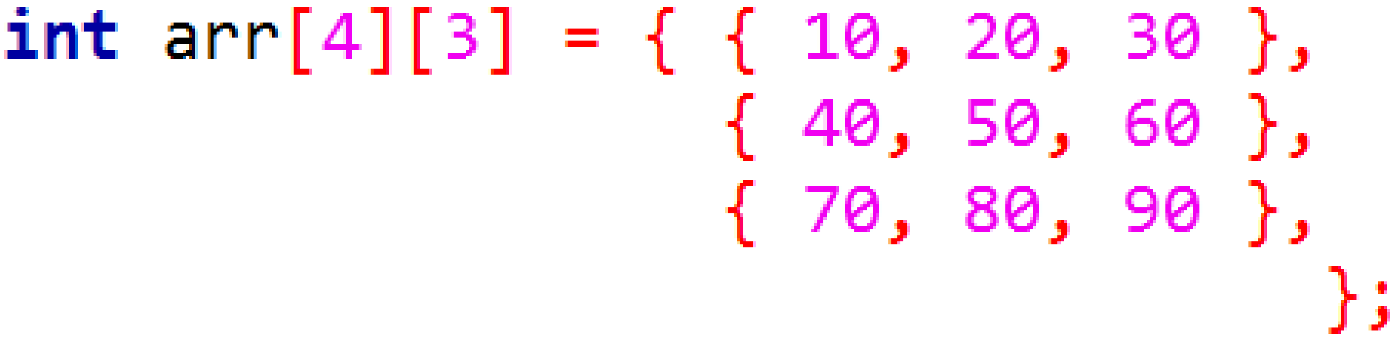
Column

Row

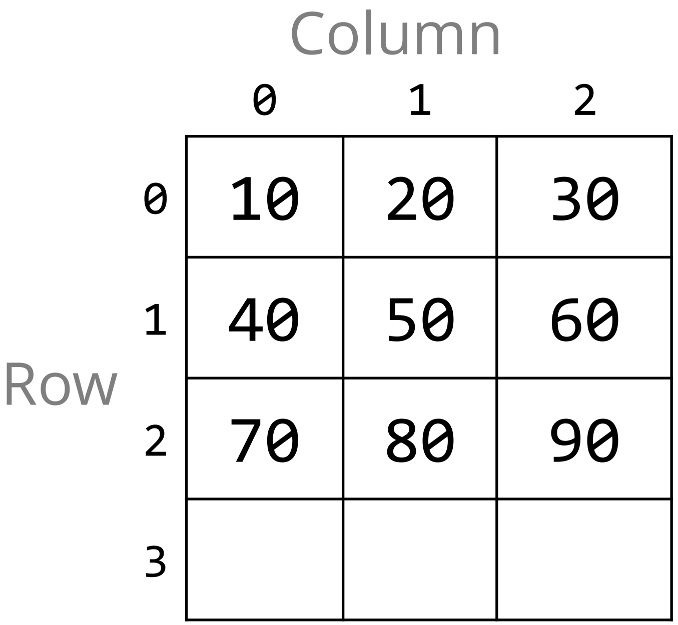
	0	1	2
0	10	20	30
1	40	50	60
2			
3			

2-D Array

Initialization



```
int arr[4][3] = { { 10, 20, 30 },
                  { 40, 50, 60 },
                  { 70, 80, 90 },
                                 };
```



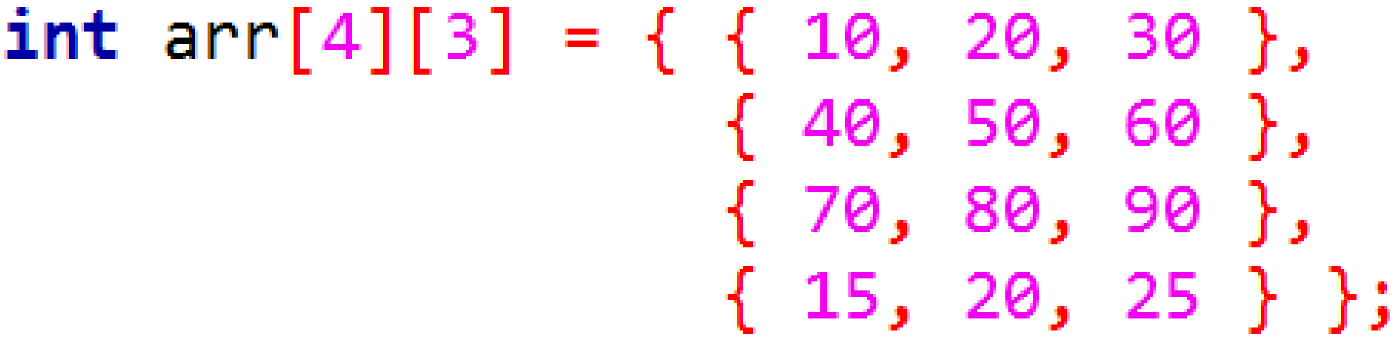
Column

Row

	0	1	2
0	10	20	30
1	40	50	60
2	70	80	90
3			

2-D Array

Initialization



```
int arr[4][3] = { { 10, 20, 30 },
                  { 40, 50, 60 },
                  { 70, 80, 90 },
                  { 15, 20, 25 } };
```

Column

Row

	0	1	2
0	10	20	30
1	40	50	60
2	70	80	90
3	15	20	25

Task 1

Initialize the following matrix. Multiply all by 10. Print it.

Column

		0	1	2
	0	1	2	3
Row	1	4	5	6
	2	7	8	9

Task 2

Initialize the following two matrix. Add them. Print the resultant matrix.

Column

		0	1	2	3
	0	1	2	3	4
Row	1	5	6	7	8
	2	9	10	11	12

Column

		0	1	2	3
	0	12	11	10	9
Row	1	8	7	6	5
	2	4	3	2	1

Task 3

Multiply the following two matrices. Print the resultant matrix.

$$\begin{array}{cc|c|c|c|c|} & & \multicolumn{4}{c}{\text{Column}} \\ & & 0 & 1 & 2 & 3 \\ \hline & 0 & 1 & 2 & 3 & 4 \\ \hline \text{Row} & 1 & 5 & 6 & 7 & 8 \\ \hline & 2 & 9 & 10 & 11 & 12 \\ \hline \end{array}$$

$$\begin{array}{cc|c|c|c|} & & \multicolumn{3}{c}{\text{Column}} \\ & & 0 & 1 & 2 \\ \hline & 0 & 10 & 20 & 30 \\ \hline & 1 & 40 & 50 & 60 \\ \hline \text{Row} & 2 & 70 & 80 & 90 \\ \hline & 3 & 15 & 20 & 25 \\ \hline \end{array}$$