AP-CSA 2D-Arrays

清森学校
BEIJING QINGSEN SCHOOL

YING HUANG

DEC.2022

INTRODUCTION TO 2D ARRAYS

清森学校
BEIJING QINGSEN SCHOOL

2D ARRAYS

- Arrays in Java can store many items of the same type.
- We can even store items in two-dimensional (2D) arrays which are arrays that have both rows and columns.
  - A **row** has horizontal elements.
  - A **column** has vertical elements.

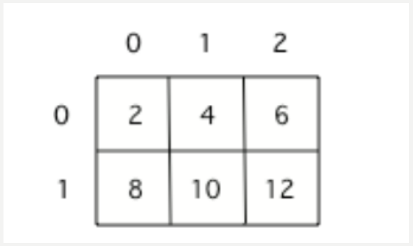
清森学校
BEIJING QINGSEN SCHOOL

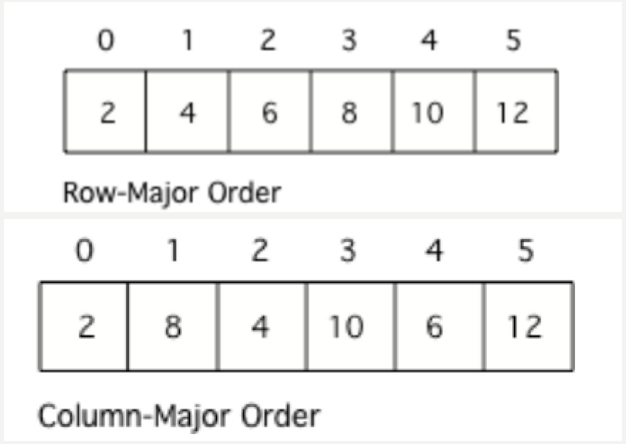
2D ARRAYS
ARRAY STORAGE

Many programming languages actually store two-dimensional array data in a one-dimensional array.

The typical way to do this is to store all the data for the first row followed by all the data for the second row and so on. This is called **row-major order**. **(Traverses a 2D array across each row)**

Some languages store all the data for the first column followed by all the data for the second column and so on. This called **column-major order. (Traverses a 2D array down each row)**





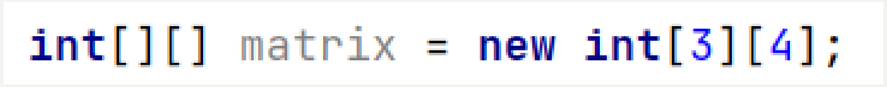
DECLARING 2D ARRAYS

To declare **and** initialize a 2D array,

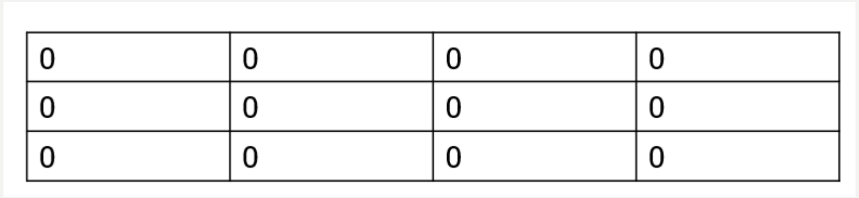
- Syntax:

```
type[][] name  //2D array of ints, null reference
type[][] name = new type[row][col];
```

- Where **row, col** is the number of rows/columns.
- When arrays are created their contents are automatically initialized to
  - 0 for numeric types,
  - null for object references,
  - false for type boolean.
- Example:



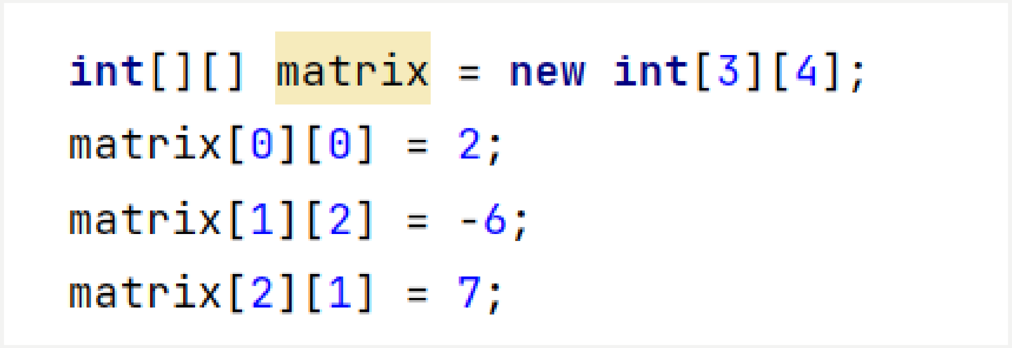
```
int[][] matrix = new int[3][4];
```



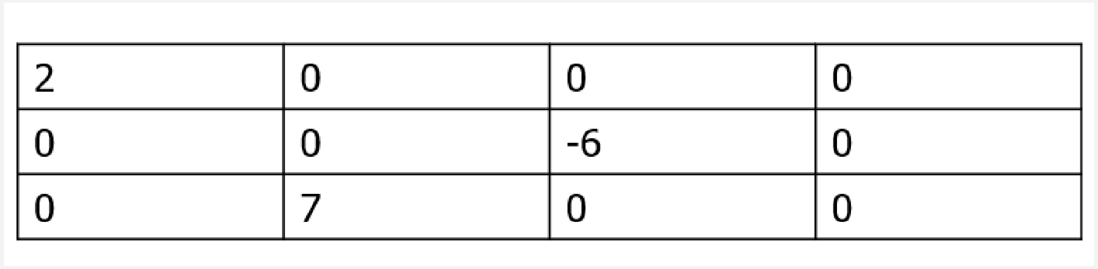
0	0	0	0
0	0	0	0
0	0	0	0

SET VALUE(S) IN A 2D ARRAY

- To explicitly put a value in an array, you can use assignment statements specifying the row and column of the entry.
- Example:



```
int[][] matrix = new int[3][4];
matrix[0][0] = 2;
matrix[1][2] = -6;
matrix[2][1] = 7;
```

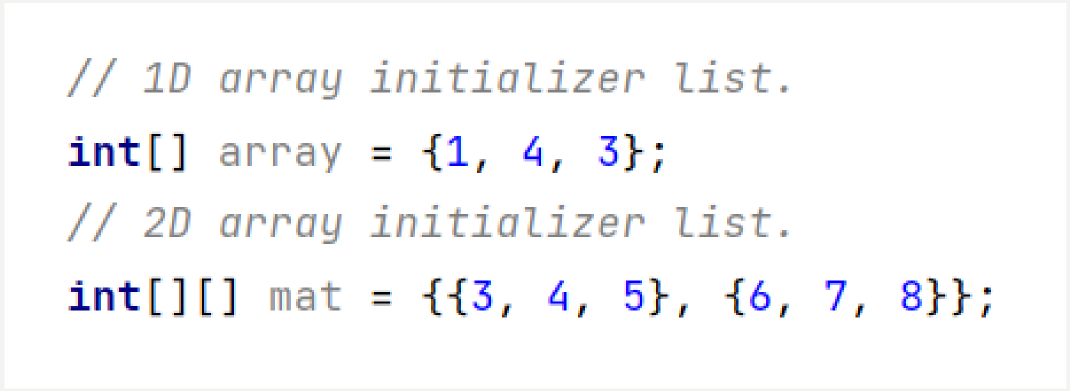


2	0	0	0
0	0	-6	0
0	7	0	0

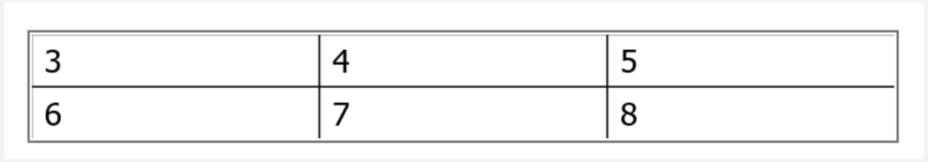
清森学校
BEIJING QINGSEN SCHOOL

INITIALIZER LIST

- You can also initialize (set) the values for the array when you create it.
- In this case you don’t need to specify the size of the array, it will be determined from the values you give. This is called using **initializer list**.
- Example:



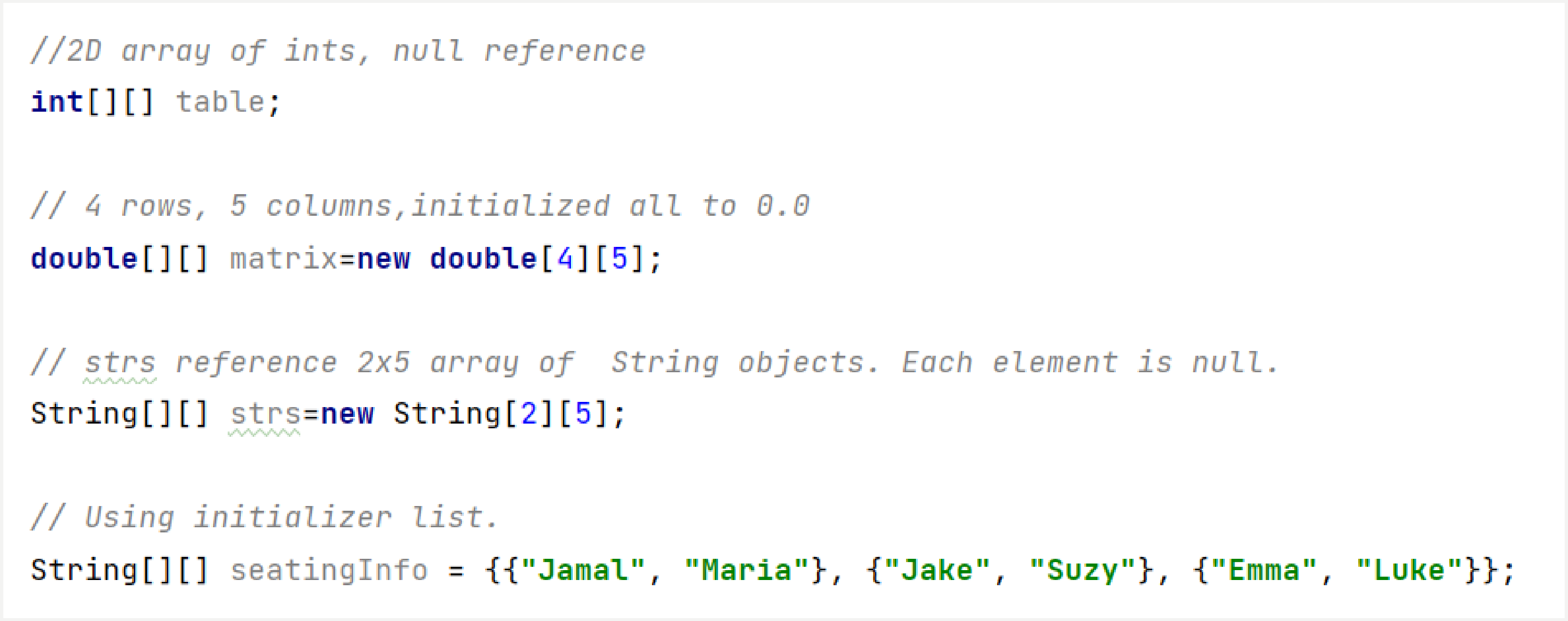
```java
// 1D array initializer list.
int[] array = {1, 4, 3};
// 2D array initializer list.
int[][] mat = {{3, 4, 5}, {6, 7, 8}};
```



3	4	5
6	7	8

DECLARE AND INITIALIZE

Summary:



```
//2D array of ints, null reference
int[][] table;

// 4 rows, 5 columns,initialized all to 0.0
double[][] matrix=new double[4][5];

// strs reference 2x5 array of  String objects. Each element is null.
String[][] strs=new String[2][5];

// Using initializer list.
String[][] seatingInfo = {{"Jamal", "Maria"}, {"Jake", "Suzy"}, {"Emma", "Luke"}};
```

ARRAY OF ARRAYS

A 2D array is implemented as an array of row arrays. Each row is a one-dimensional array of elements. Suppose that mat is the 2D array:

3	-4	1	2
6	0	8	1
-2	9	1	7

- Then mat is an array of three arrays:
- **mat[0]** is the one-dimensional array {3,-4,1,2}.
- **mat[1]** is the one-dimensional array {6,0,8,1}.
- **mat[2]** is the one-dimensional array {-2,9,1,7}.

清森学校
BEIJING QINGSEN SCHOOL

ARRAY OF ARRAYS

3	-4	1	2
6	0	8	1
-2	9	1	7

- **mat.length** is the number of rows.
  - In this case, it equals 3 because there are three row-arrays in mat.

- For each row k , where $\mathbf{0 \leq k < mat.length}$, **mat[k].length** is the number of elements in that row, namely the number of columns.
  - In this case, mat[k].length=4 for all k.

- Java allows "jagged arrays" where each row array may have different lengths.
  - **However, on the AP exam, assume all arrays are rectangular.**

清森学校
BEIJING QINGSEN SCHOOL

TRAVERSE 2D-ARRAYS

COMMON ALGORITHMS

You should know how to implement the following algorithms:

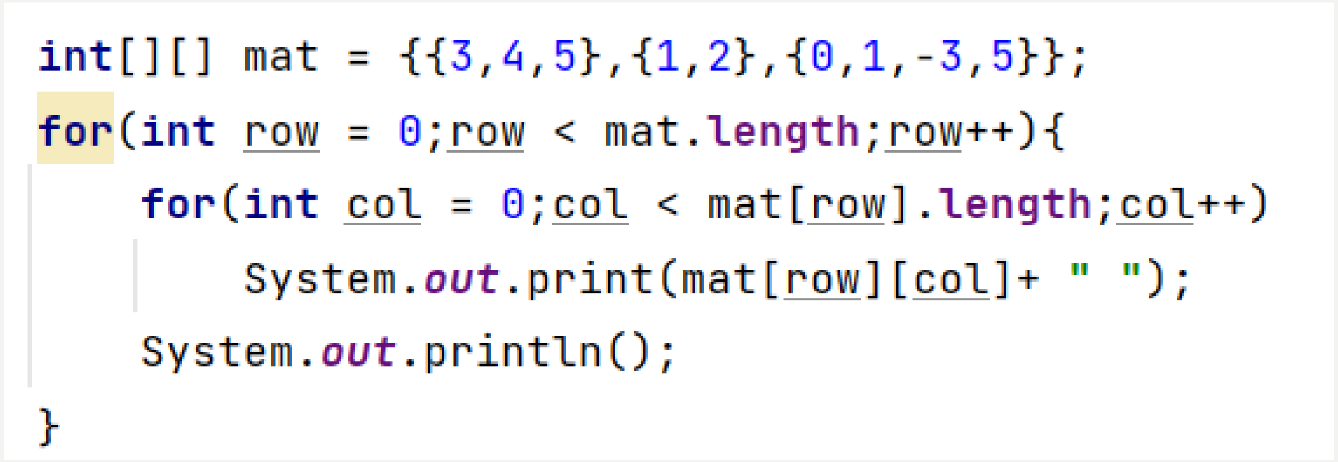
Given a 2D array:

- Traverse the array by row major order
- Traverse the array by column major order
- Traverse one row of the 2D array
- Traverse one column of the 2D array
- Traverse row-by-row
- Find the largest element.
- Find the sum and average.
- Sequential/Linear search a 2D array by sequential/search each row of 2D array.

清森学校
BEIJING QINGSEN SCHOOL

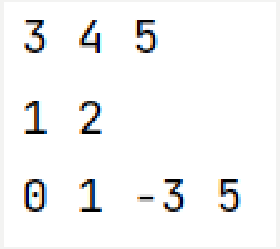
ROW MAJOR ORDER

- Suppose that mat is a 2D array initialized with integers. Use nested for loop to print out the elements of the array. Traverse by **row-major order**.
- Example Code:



```
int[][] mat = {{3,4,5},{1,2},{0,1,-3,5}};
for(int row = 0;row < mat.length;row++){
    for(int col = 0;col < mat[row].length;col++)
        System.out.print(mat[row][col]+ " ");
    System.out.println();
}
```

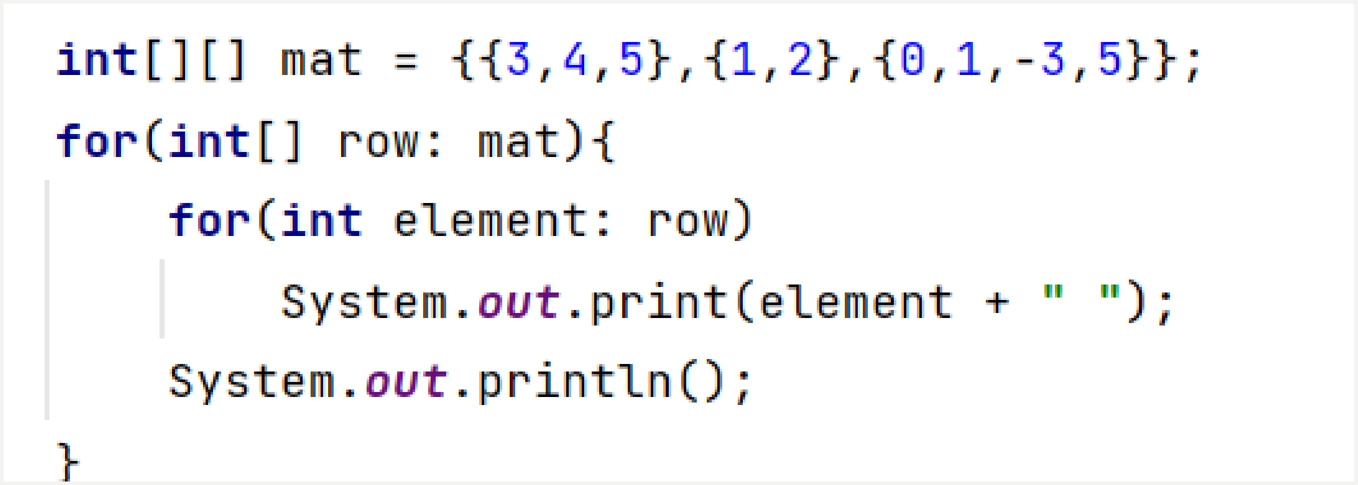
- Output:



```
3 4 5
1 2
0 1 -3 5
```

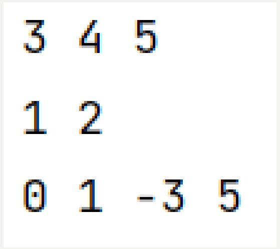
FOR EACH TRAVERSAL

- Traverse an array by using a for each loop. For each loop, in general, are much easier to work with. If you are not modifying your 2D array, it is highly recommended that you use for each to avoid index errors.
- Example Code:



```java
int[][] mat = {{3,4,5},{1,2},{0,1,-3,5}};
for(int[] row: mat){
    for(int element: row)
        System.out.print(element + " ");
    System.out.println();
}
```

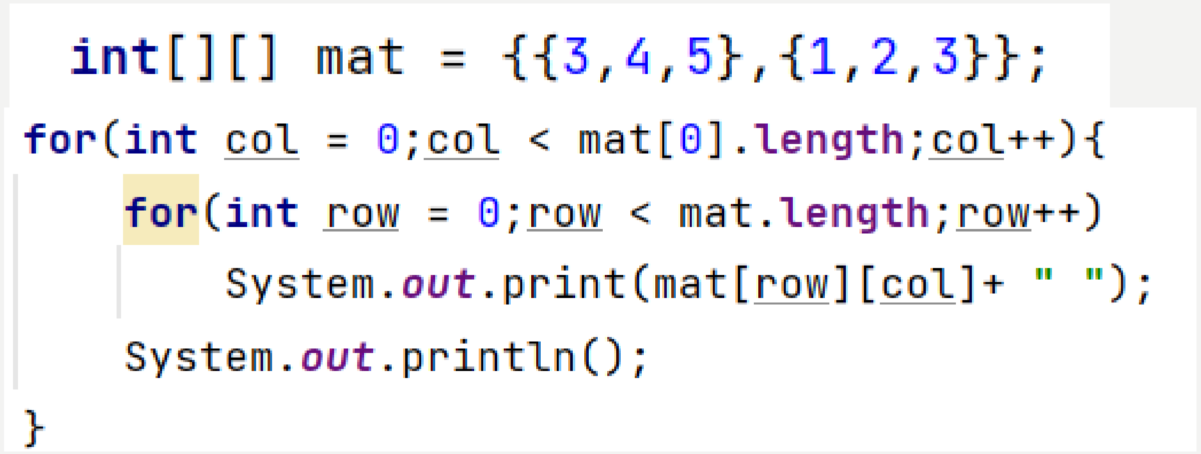
- Output:



```
3 4 5
1 2
0 1 -3 5
```

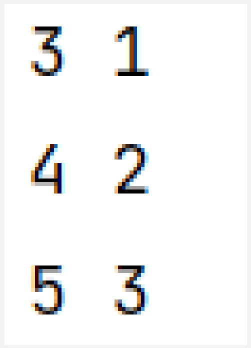
COLUMN MAJOR ORDER

- Suppose that mat is a 2D array initialized with integers. Use nested for loop to print out the elements of the array. Traverse by **column-major order**. Assume that the array is rectangular.
- Example Code:



```java
int[][] mat = {{3,4,5},{1,2,3}};
for(int col = 0;col < mat[0].length;col++){
    for(int row = 0;row < mat.length;row++)
        System.out.print(mat[row][col]+ " ");
    System.out.println();
}
```

- Output:

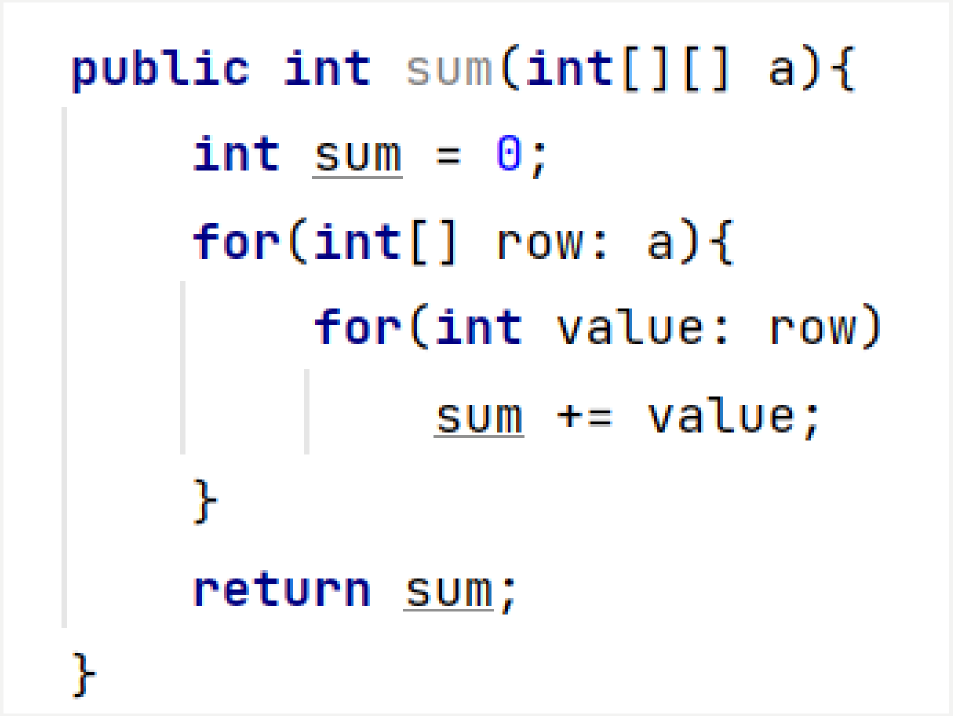


```
3 1
4 2
5 3
```

SUM

Write the method that returns the sum of a 2D array.

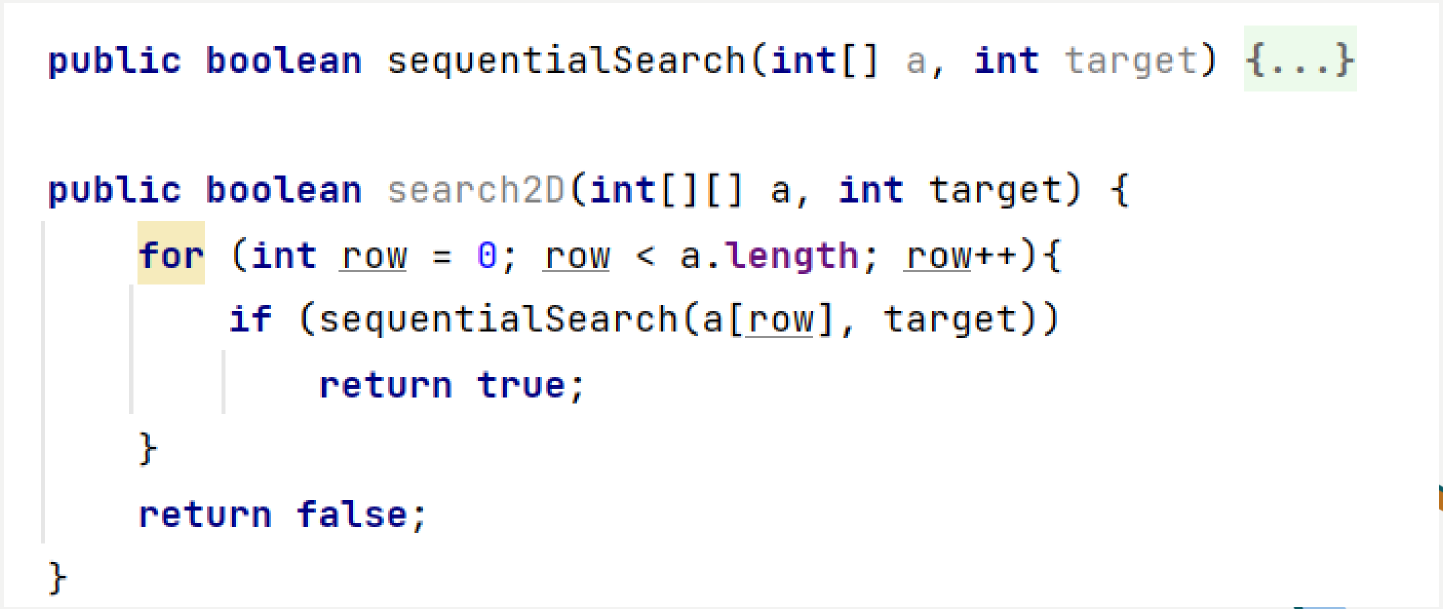
Example :



```java
public int sum(int[][] a){
    int sum = 0;
    for(int[] row: a){
        for(int value: row)
            sum += value;
    }
    return sum;
}
```

SEARCHING AN ARRAY

- Write the method that searches a 2D array for a target value. Returns true if target is in the 2D array and false otherwise. Assume the sequentialSearch for 1D below has already been implemented. (see previous lecture).
- Example Code:



```
public boolean sequentialSearch(int[] a, int target) {...}

public boolean search2D(int[][] a, int target) {
    for (int row = 0; row < a.length; row++){
        if (sequentialSearch(a[row], target))
            return true;
    }
    return false;
}
```

2D ARRAYS OF OBJECTS

```
Point[][] pointMatrix;
```

Suppose that pointMatrix is initialized with Point objects. Change the x-coordinate of each Point to 1. Suppose that instance variables are private. Use the mutator method setX().

```
for(int row = 0;row < pointMatrix.length;row++)
  for(int col = 0;col < pointMatrix[0].length;col++)
     pointMatrix[row][col].setX(1);
```

We can do this more simply with a for each loop! See next slide.

2D ARRAYS OF OBJECTS

We can do the previous problem more simply with a for each loop.

Compare the two methods:

Regular for Loop:

```
for(int row = 0;row < pointMatrix.length;row++)
  for(int col = 0;col < pointMatrix[0].length;col+
     pointMatrix[row][col].setX(1);
```

For Each Loop:

```
for(Point[] row: pointMatrix){
  for(Point pt: row)
    pt.setX(1);
```

2D ARRAYS

Example 1

How many columns does a have if it is created as follows int[][] a = { {2, 4, 6, 8}, {1, 2, 3, 4} };?

A. 2

B. 4

C. 8

ANS: B

清森学校
BEIJING QINGSEN
SCHOOL

2D ARRAYS

Example 2

How would you get the value 6 out of the following array int[][] a = { {2, 4, 6, 8}, {1, 2, 3, 4} };?

A. a[0][3]

B. a[1][3]

C. a[0][2]

D. a[2][0]

ANS: C

2D ARRAYS

Example 3

Given the following code segment, what is the value of sum after this code executes?

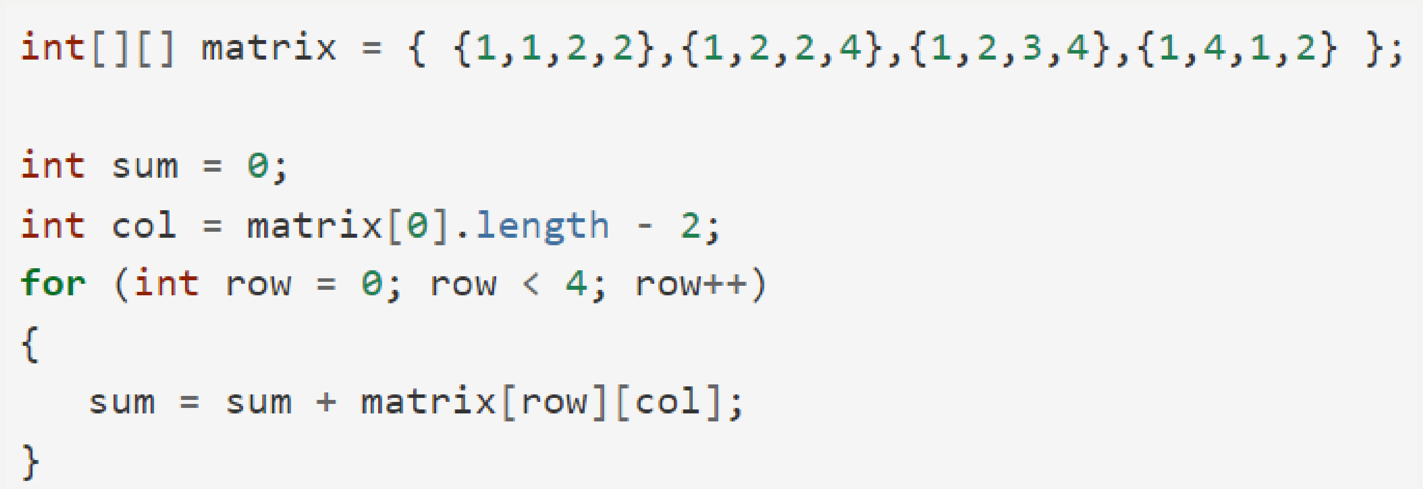
A. 4

B. 8

C. 9

D. 12

ANS: B



```
int[][] matrix = { {1,1,2,2},{1,2,2,4},{1,2,3,4},{1,4,1,2} };

int sum = 0;
int col = matrix[0].length - 2;
for (int row = 0; row < 4; row++)
{
   sum = sum + matrix[row][col];
}
```

2D ARRAYS

Example 4

What are the contents of mat after the following code segment has been executed?

```java
int [][] mat = new int [4][3];
for (int row = 0; row < mat.length; row++) {
   for (int col = 0; col < mat[0].length; col++) {
      if (row < col)
         mat[row][col] = 1;
      else if (row == col)
         mat[row][col] = 2;
      else
         mat[row][col] = 3; } }
```

A. { {2 3 3}, {1 2 3}, {1 1 2}, {1 1 1} }

B. { {2 1 1}, {3 2 1}, {3 3 2}, {3 3 3} }

C. { {2 1 1 1}, {3 2 1 1}, {3 3 2 1} }

D. { {2 3 3 3}, {1 2 3 3}, {1 1 2 3} }

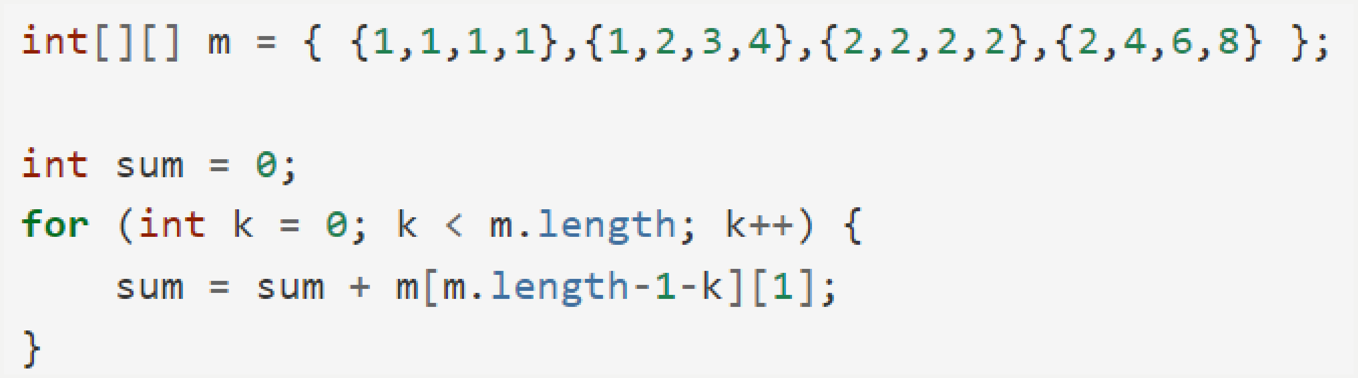
ANS: B

清森学校
BEIJING QINGSEN
SCHOOL

2D ARRAYS

Example 5

Given the following code segment, what is the value of sum after this code executes?



```
int[][] m = { {1,1,1,1},{1,2,3,4},{2,2,2,2},{2,4,6,8} };

int sum = 0;
for (int k = 0; k < m.length; k++) {
    sum = sum + m[m.length-1-k][1];
}
```

A. 4

B. 6

C. 9

D. 10

ANS: C

2D ARRAYS

Example 6

What are the contents of arr after the following code has been executed?

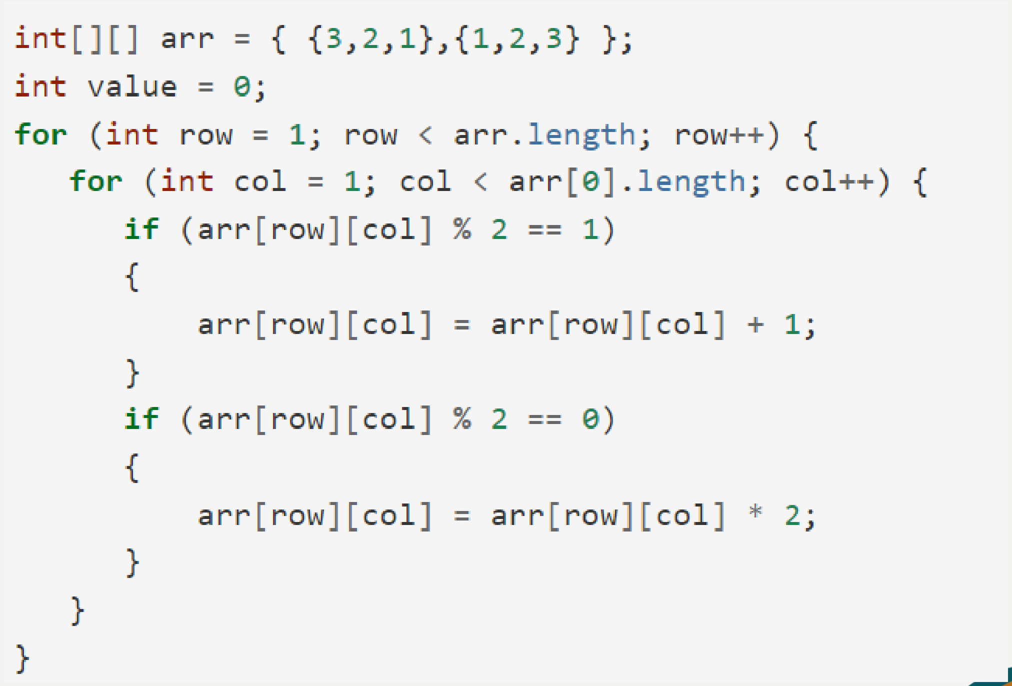
A. { {6, 4, 2}, {2, 4, 6} }

B. { {3, 2, 1}, {1, 4, 6} }

C. { {3, 2, 1}, {1, 4, 8} }

D. { {4, 4, 2}, {2, 4, 4} }

ANS: C



```
int[][] arr = { {3,2,1},{1,2,3} };
int value = 0;
for (int row = 1; row < arr.length; row++) {
   for (int col = 1; col < arr[0].length; col++) {
      if (arr[row][col] % 2 == 1)
      {
          arr[row][col] = arr[row][col] + 1;
      }
      if (arr[row][col] % 2 == 0)
      {
          arr[row][col] = arr[row][col] * 2;
      }
   }
}
```

清森学校
BEIJING QINGSEN SCHOOL