**2 Ejercicios Arrays multidimensionales**

**En todos los ejercicios mostrar el array solución.**

1.- Invertir una matriz de modo que todos los elementos de la derecha pasen a estar a la izquierda y viceversa. Mostrar las dos matrices.

Probarlo con:

{{1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12}, {13, 14, 15, 16}}

**int** a[][]={{1, 2, 3, 4},{5, 6, 7, 8},{9, 10, 11, 12},{13, 14, 15, 16}};

**int** sol[][]=**new** **int** [a.length][a[0].length];

**int** posI=0;

**int** posJ=0;

**for** (**int** i=a.length-1;i>=0;i--) {

**for** (**int** j=a[i].length-1; j>=0;j--) {

sol[posI][posJ]=a[i][j];

posJ++;

}

posI++;

posJ=0;

}

System.***out***.println(Arrays.*deepToString*(sol));

2.- Invertir una matriz de modo que todos los elementos de la abajo pasen a estar a arriba y viceversa. Mostrar las dos matrices.

Probarlo con:

{{1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12}, {13, 14, 15, 16}}

**int** a[][]={ //fila

{1, 2, 3, 4}, // 0

{5, 6, 7, 8}, // 1

{9, 10, 11, 12}, // 2

{13, 14, 15, 16}};//3

**int** sol[][]=**new** **int** [a.length][a[0].length];

**int** posI=a.length-1;

**int** posJ=0;

**for** (**int** i=0; i<sol.length;i++) {

**for** (**int** j=0; j<sol[i].length;j++) {

sol[i][j]=a[posI][posJ];

posJ++;

}

posI--;

posJ=0;

}

System.***out***.println(Arrays.*deepToString*(sol));

System.***out***.println(Arrays.*deepToString*(a));

3.- Crear un array bidimensional que contenga caracteres y donde cada fila contenga los caracteres de cada palabra de una frase. Probarlo con:

“Esta es la frase que meteremos en la matriz”

String frase="Esta es la frase que meteremos en la matriz";

// el nº de filas= nº de palabras

String[] palabras=frase.split(" ");

**char** carac[][]=**new** **char**[palabras.length][];

**for** (**int** i=0; i<palabras.length;i++) {

carac[i]=**new** **char** [palabras[i].length()];

}

**for** (**int** i=0;i<palabras.length;i++) {

**for** (**int** j=0;j<palabras[i].length();j++) {

carac[i][j]=palabras[i].charAt(j);

}

}

System.***out***.println(Arrays.*deepToString*(carac));

4.- Invertir una matriz tomando como eje su diagonal (la diagonal que va de i=0, j=0 a j=n, i=n). Mostrar las dos matrices. Probarlo con:

{{2, 5, 6, 1, 6}, {7, 3, 2, 8, 0}, {4, 9, 2, 10, 7}, {1, 8, 3, 2, 6}, {9, 2, 6, 1, 3}}

**int** mat[][]={

{2, 5, 6, 1, 6},

{7, 3, 2, 8, 0},

{4, 9, 2, 10, 7},

{1, 8, 3, 2, 6},

{9, 2, 6, 1, 3}};

// 00 01 02 03

// 10 11 12 13

// 20 21 22 23

// 30 31 32 33

**int** sol[][]=**new** **int** [mat.length][mat[0].length];

**for** (**int** i=0;i<mat.length;i++) {

**for** (**int** j=0;j<mat[i].length;j++) {

sol [i][j]=mat[j][i];

}

}

System.***out***.println(Arrays.*deepToString*(sol));

System.***out***.println(Arrays.*deepToString*(mat));

5.- A partir de dos arrays bidimensionales con el mismo número de filas pero donde éstas puedes ser de diferente tamaño obtener un tercero donde ponga:

1: si el elemento mayor está en la primera matriz.

2: si el elemento mayor está en la segunda matriz

0: si son iguales

-1: si el elemento solo existe en la matriz 1

-2: si el elemento solo existe en la matriz 2

Probarlo con:

{{23, 4, 5, 1, 9, 8}, {43, 12, 67, 7}, {2, 71}, {9, 1, 6, 14}} y

{{11, 6, 5, 9, 1}, {9, 12, 8, 1}, {6, 9, 3, 2}, {4, 1, 6, 8}}

**int** a[][]={{23, 4, 5, 1, 9, 8}, {43, 12, 67, 7}, {2, 71}, {9, 1, 6, 14}} ;

**int** b[][]={{11, 6, 5, 9, 1}, {9, 12, 8, 1}, {6, 9, 3, 2}, {4, 1, 6, 8}};

**int** sol[][]= **new** **int** [a.length][];

//Crear array solucion

**for** (**int** i=0;i<a.length;i++) {

**if** (a[i].length>b[i].length) {

sol [i]=**new** **int** [a[i].length];

}

**else** {

sol [i]=**new** **int** [b[i].length];

}

}

//rellenar array solucion

**for** (**int** i=0;i<a.length;i++) {

**if** (a[i].length>b[i].length) {

//en caso de que la fila de a sea mayor

**int** j=0;

**for** (j=0;j<b[i].length;j++) {

**if** (a[i][j]>b[i][j]) {

sol[i][j]=1;}

**else** **if**(a[i][j]<b[i][j]) {

sol[i][j]=2;}

**else** {

sol[i][j]=0;}

}

//rellenar los siguinetes con -1

**for** (**int** k=j;k<a.length;k++) {

sol [i][k]=-1;

}

}

//en caso de que la fila de b sea mayor

**else** **if** (a[i].length<b[i].length) {

**for** (**int** j=0;j<a[i].length;j++) {

**if** (a[i][j]>b[i][j]) {

sol[i][j]=1;

}

**else** **if**(a[i][j]<b[i][j]) {

sol[i][j]=2;

}

**else** {

sol[i][j]=0;

}

}

}

**else** {

**for** (**int** j=0;j<b[i].length;j++) {

**if** (a[i][j]>b[i][j]) {

sol[i][j]=1;

}

**else** **if**(a[i][j]<b[i][j]) {

sol[i][j]=2;

}

**else** {

sol[i][j]=0;

}

}

}

}

System.***out***.println(Arrays.*deepToString*(sol));

6.- En un  array bidimensional dejar en las filas pares solo los elementos con valor par y en las filas impares los elementos con valor impar. Del nuevo array se eliminarán los elementos que sobran, NO pondrá 0 en su lugar. Si no hay ningún elemento que cumpla la condición en esa fila pondrá -1.

Probarlo con:

{{2, 3, 4, 5, 6, 7}, {9, 10, 2, 3, 5}, { 9, 14, 17, 12, 6} , {4, 8, 10}, {6, 7, 4, 9, 10}, {1, 2}, {8, 9, 10, 11}}

**int** a[][]={{2, 3, 4, 5, 6, 7}, {9, 10, 2, 3, 5}, { 9, 14, 17, 12, 6} , {4, 8, 10}, {6, 7, 4, 9, 10}, {1, 2}, {8, 9, 10, 11}};

**int** sol[][]=**new** **int** [a.length][];

**for** (**int** i=0;i<a.length;i++) {

**int** cont=0;

**for** (**int** j=0;j<a[i].length;j++) {

**if** (i%2==0 && a[i][j]%2==0) {

cont++;

}

**else** **if** (i%2!=0 && a[i][j]%2!=0){

}

}

//System.out.println("fila"+i+" valores que cumplen: "+cont);

**if** (cont ==0) {

sol[i]=**new** **int** [1];// defino uno que tenga por lo menos 1

sol[i][0]=-1;

}

**else** {

sol[i]=**new** **int** [cont];//es del tamaño de lo que pone en el contador

**int** contSol=0; //creamos otro contador para recorrer esa fila

**for** (**int** j=0;j<a[i].length;j++) {

**if** (i%2==0 && a[i][j]==0) {

sol[i][contSol]=a[i][j];

contSol++;

}

**else** **if** (i%2!=0 && a[i][j]%2!=0) {

sol[i][contSol]=a[i][j];

contSol++;

}

}

}

}

System.***out***.println(Arrays.*deepToString*(sol));

7.- Dado un array bidimensional obtener en qué fila y en qué columna está el elemento más grande, y qué valor tiene. Probarlo con:

{{45, 9, 12, 2,0}, {18, 7, 86, 9, 2, 7, 9},{1, 7, 34, 2, 8},{3, 4, 5}}

**int** a[][]={{45, 9, 12, 2,0}, {18, 7, 86, 9, 2, 7, 9},{1, 7, 34, 2, 8},{3, 4, 5}};

**int** mayor=0;

**int** posI=0;

**int** posJ=0;

**for** (**int** i=0;i<a.length;i++) {

**for** (**int** j=0;j<a[i].length;j++) {

**int** num=a[i][j];

**if**(num>mayor) {

mayor=num;

posI=i;

posJ=j;

}

}

}

System.***out***.println("El nº mayor es : "+ mayor+" y esta en la posicion i= "+posI+" ,j= "+posJ);

8.- Obtener los 3 elementos más grandes de una matriz donde no hay ningún número repetido. Probarlo con:

{{23, 5, 2, 9, 4}, {8, 76, 34, 10, 1}, {29, 0, 7, 14, 3},

{38, 6, 18, 52, 47}}

**int** a[][]={{23, 5, 2, 9, 4}, {8, 76, 34, 10, 1}, {29, 0, 7, 14, 3}, {38, 6, 18, 52, 47}};

**int** mayorMax=0;

**int** mayorMin=0;

**int** mayorInt=0;

**int** posI=0;

**int** posJ=0;

**for** (**int** i=0; i<a.length;i++) {

**for**(**int** j=0;j<a[i].length;j++) {

**int** num=a[i][j];

**if**(num>mayorMax) {

mayorMax=num;

}

num=a[i][j];

**if** (num>mayorInt && num<mayorMax) {

mayorInt=num;

}

num=a[i][j];

**if**(num>mayorMin && num<mayorInt) {

mayorMin=num;

}

}

}

System.***out***.println("Los tres elementos más grandes de mayro a menos son: "+ mayorMax+","+mayorInt+" y "+mayorMin);

9.- Ordenar los valores de una matriz 3x4 de menor a mayor y de arriba abajo y de izquierda a derecha. Probarlo con:

{{9, -7, 5, 3}, {12, 23, -2, 8}, {4, 7, 9, 0}}

**int** a[][]={{9, -7, 5, 3}, {12, 23, -2, 8}, {4, 7, 9, 0}};

// DE MENOR A MAYOR

**int** b[]= **new** **int** [a.length\*a[0].length];

**int** count=0;

**for** (**int** i=0;i<a.length;i++) {

**for** (**int** j=0;j<a[i].length;j++) {

b[count]=a[i][j];

count++;

}

}

Arrays.*sort*(b);

System.***out***.println(Arrays.*toString*(b));

count=0;

**int** c[][]=**new** **int** [a.length][a[0].length];

**for** (**int** i=0;i<a.length;i++) {

**for**(**int** j=0;j<a[i].length;j++) {

c[i][j]=b[count];

count++;

}

}

System.***out***.println(Arrays.*deepToString*(c));

// DE ARRIBA ABAJO

**int** d[][]=**new** **int** [c.length][c[0].length];

**int** posI=c.length-1;

**int** posJ=0;

**for** (**int** i=0; i<d.length;i++) {

**for** (**int** j=0; j<d[i].length;j++) {

d[i][j]=c[posI][posJ];

posJ++;

}

posI--;

posJ=0;

}

System.***out***.println(Arrays.*deepToString*(d));

//DE DERECHA A IZQUIERDA

**int** e[][]=**new** **int** [d.length][d[0].length];

posI=0;

posJ=0;

**for** (**int** i=d.length-1;i>=0;i--) {

**for** (**int** j=d[i].length-1; j>=0;j--) {

e[posI][posJ]=d[i][j];

posJ++;

}

posI++;

posJ=0;

}

System.***out***.println(Arrays.*deepToString*(e));

10.- Recolocar las filas de una array bidimensional de modo que el nuevo vaya de las filas más cortas a las más largas. Las filas han de mantener sus valores. Probarlo con:

{{1, 2, 4, 6, 4}, {9, 7}, {2, 6, 8, 3}, {7,6, 9}}

**int** a[][]={

{1, 2, 4, 6, 4},

{9, 7},

{2, 6, 8, 3},

{7,6, 9}

};

**for** (**int** i=0;i<a.length;i++) {

**for** (**int** j=i+1;j<a.length;j++) {

**if** (a[j].length<a[i].length) {

**int** temp[]=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

System.***out***.println(Arrays.*deepToString*(a));