Contenido

[**Sección 13: Matrices: arreglos bidimensionales** 2](#_Toc139991563)

[116. Introducción a las matrices 2](#_Toc139991564)

[117. Ejemplo de una matriz de enteros 2](#_Toc139991565)

[118. Ejemplo de un arreglo de string e iterando con for 2](#_Toc139991566)

[119. Tamaños de columnas variable en una matriz 2](#_Toc139991567)

[120. Buscando un elemento en una matriz usando etiquetas y break 2](#_Toc139991568)

[121. Ejecutando en modo depuración 2](#_Toc139991569)

[122. Matriz simétrica 2](#_Toc139991570)

[123. Convertir a la matriz transpuesta 2](#_Toc139991571)

[124. Creando la matriz transpuesta de una matriz 2](#_Toc139991572)

[125. Operaciones aritméticas en matrices 2](#_Toc139991573)

[126. Sumar filas y columnas de una matriz 2](#_Toc139991574)

[127. Límites inferior y superior de la matriz y la diagonal principal 2](#_Toc139991575)

[128. Descargar Código Fuente 2](#_Toc139991576)

[Tarea 18: Imprimir una X 2](#_Toc139991577)

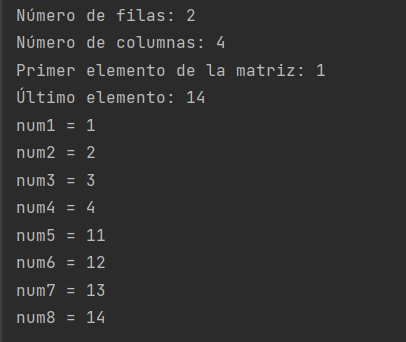
[Tarea 19: Imprimir un silla 2](#_Toc139991578)

# **Sección 13: Matrices: arreglos bidimensionales**

# 116. Introducción a las matrices

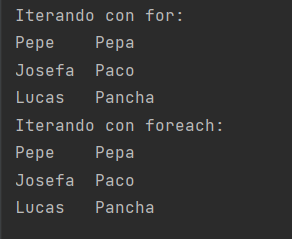
# 117. Ejemplo de una matriz de enteros

public class EjemploMatrices {  
 public static void main(String[] args) {  
  
 // Crear una matriz de tamaño 2x4  
 int[][] numeros = new int[2][4];  
  
 // Asignar valores a los elementos de la matriz  
 numeros[0][0] = 1;  
 numeros[0][1] = 2;  
 numeros[0][2] = 3;  
 numeros[0][3] = 4;  
  
 numeros[1][0] = 11;  
 numeros[1][1] = 12;  
 numeros[1][2] = 13;  
 numeros[1][3] = 14;  
  
 // Obtener el número de filas y columnas de la matriz  
 System.*out*.println("Número de filas: " + numeros.length);  
 System.*out*.println("Número de columnas: " + numeros[0].length);  
  
 // Acceder al primer elemento de la matriz  
 System.*out*.println("Primer elemento de la matriz: " + numeros[0][0]);  
  
 // Acceder al último elemento de la matriz  
 System.*out*.println("Último elemento: " + numeros[numeros.length - 1][numeros[1].length - 1]);  
  
 // Asignar valores individuales a variables separadas  
 int num1 = numeros[0][0];  
 int num2 = numeros[0][1];  
 int num3 = numeros[0][2];  
 int num4 = numeros[0][3];  
 int num5 = numeros[1][0];  
 int num6 = numeros[1][1];  
 int num7 = numeros[1][2];  
 int num8 = numeros[1][3];  
  
 // Imprimir los valores asignados a las variables  
 System.*out*.println("num1 = " + num1);  
 System.*out*.println("num2 = " + num2);  
 System.*out*.println("num3 = " + num3);  
 System.*out*.println("num4 = " + num4);  
 System.*out*.println("num5 = " + num5);  
 System.*out*.println("num6 = " + num6);  
 System.*out*.println("num7 = " + num7);  
 System.*out*.println("num8 = " + num8);  
 }  
}

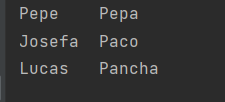


# 118. Ejemplo de un arreglo de string e iterando con for

public class EjemploMatricesStringFor {  
 public static void main(String[] args) {  
  
 // Crear una matriz de tipo String de tamaño 3x2  
 String[][] nombres = new String[3][2];  
  
 // Asignar valores a los elementos de la matriz  
 nombres[0][0] = "Pepe";  
 nombres[0][1] = "Pepa";  
 nombres[1][0] = "Josefa";  
 nombres[1][1] = "Paco";  
 nombres[2][0] = "Lucas";  
 nombres[2][1] = "Pancha";  
  
 // Iterar sobre la matriz utilizando un bucle "for"  
 System.*out*.println("Iterando con for:");  
 for(int i = 0; i < nombres.length; i++){  
 for(int j = 0; j < nombres[i].length; j++){  
 System.*out*.print(nombres[i][j] + "\t");  
 }  
 System.*out*.println();  
 }  
  
 // Iterar sobre la matriz utilizando un bucle "foreach"  
 System.*out*.println("Iterando con foreach:");  
 for(String[] fila: nombres){  
 for(String nombre: fila){  
 System.*out*.print(nombre + "\t");  
 }  
 System.*out*.println();  
 }  
 }  
}

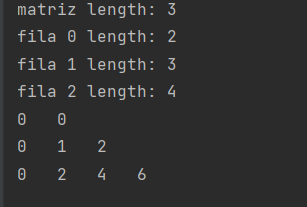


public class EjemploMatricesStringFor2 {  
 public static void main(String[] args) {  
  
 // Crear una matriz de tipo String e inicializarla con los valores proporcionados  
 String[][] nombres = { {"Pepe", "Pepa"}, {"Josefa", "Paco"}, {"Lucas", "Pancha"}};  
  
 // Iterar sobre la matriz utilizando un bucle "foreach"  
 for(String[] fila: nombres){ // Itera sobre cada fila de la matriz  
 for(String nombre: fila){ // Itera sobre cada elemento de la fila actual  
 System.*out*.print(nombre + "\t"); // Imprime el nombre actual seguido de una tabulación  
 }  
 System.*out*.println(); // Imprime una nueva línea después de imprimir todos los elementos de la fila  
 }  
 }  
}



# 119. Tamaños de columnas variable en una matriz

public class EjemploMatricesColumnasVariable {  
 public static void main(String[] args) {  
  
 // Declarar una matriz con 3 filas y columnas variables  
 int[][] matriz = new int[3][];  
  
 // Asignar tamaño a las columnas de cada fila  
 matriz[0] = new int[2];  
 matriz[1] = new int[3];  
 matriz[2] = new int[4];  
  
 // Imprimir la longitud de la matriz y de cada fila  
 System.*out*.println("matriz length: " + matriz.length);  
 System.*out*.println("fila 0 length: " + matriz[0].length);  
 System.*out*.println("fila 1 length: " + matriz[1].length);  
 System.*out*.println("fila 2 length: " + matriz[2].length);  
  
 // Asignar valores a los elementos de la matriz  
 for(int i = 0; i < matriz.length; i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 matriz[i][j] = i \* j;  
 }  
 }  
  
 // Imprimir la matriz  
 for(int i = 0; i < matriz.length; i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 System.out.print(matriz[i][j] + "\t");  
 }  
 System.out.println();  
 }  
 }  
}



# 120. Buscando un elemento en una matriz usando etiquetas y break

public class EjemploMatricesBuscar {  
 public static void main(String[] args) {  
  
 // Declarar y inicializar una matriz de enteros  
 int[][] matrizDeEnteros = {  
 {35, 90, 3, 1978},  
 {15, 2020, 10, 5},  
 {677, 127, 32767, 1999}  
 };  
  
 // Elemento a buscar en la matriz  
 int elementoBuscar = 1999;  
  
 // Variables para controlar si se encontró el elemento  
 boolean encontrado = false;  
 int i;  
 int j = 0;  
  
 // Bucle etiquetado para buscar el elemento en la matriz  
 buscar: for(i = 0; i < matrizDeEnteros.length; i++){  
 for(j = 0; j < matrizDeEnteros[i].length; j++){  
 if(matrizDeEnteros[i][j] == elementoBuscar){  
 encontrado = true;  
 break buscar; // Salir del bucle etiquetado si se encuentra el elemento  
 }  
 }  
 }  
  
 // Verificar si el elemento fue encontrado  
 if(encontrado){  
 System.*out*.println("Encontrado " + elementoBuscar + " en las coordenadas " + i + "," + j);  
 } else {  
 System.*out*.println(elementoBuscar + " no se encontró en la matriz!");  
 }  
 }  
}



# 121. Ejecutando en modo depuración

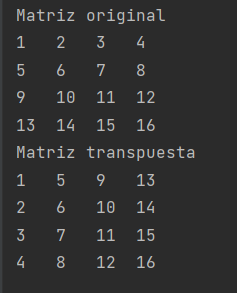
# 122. Matriz simétrica

public class EjemploMatricesSimetrica {  
 public static void main(String[] args) {  
 // Variable para controlar si la matriz es simétrica  
 boolean simetrica = true;  
  
 // Declarar y definir una matriz  
 int[][] matriz = {  
 {1,2,3,4},  
 {2,1,0,5},  
 {3,0,1,6},  
 {4,5,6,7}  
 };  
  
 // Bucle para verificar la simetría de la matriz  
 salir: for(int i = 0; i < matriz.length; i++){  
  
 for(int j = 0;j < i; j++){  
 // Comprobar si los elementos son diferentes en la simetría opuesta  
 if(matriz[i][j] != matriz[j][i]){  
 simetrica = false;  
 break salir; // Salir del bucle etiquetado si no es simétrica  
 }  
 }  
 }  
  
 // Imprimir el resultado de la verificación de simetría  
 if(simetrica){  
 System.*out*.println("La matriz es simétrica");  
 } else{  
 System.*out*.println("La matriz no es simétrica");  
 }  
 }  
}

# 

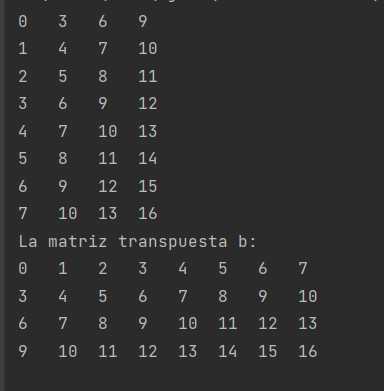
# 123. Convertir a la matriz transpuesta

public class EjemploMatricesTranspuesta {  
 public static void main(String[] args) {  
 // Declaración e inicialización de la matriz original  
 int[][] matriz = {  
 {1, 2, 3, 4},  
 {5, 6, 7, 8},  
 {9, 10, 11, 12},  
 {13, 14, 15, 16}  
 };  
  
 // Imprimir la matriz original  
 System.*out*.println("Matriz original");  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 System.*out*.print(matriz[i][j] + "\t");  
 }  
 System.*out*.println();  
 }  
  
 int aux;  
 for (int i = 1; i < matriz.length; i++) {  
 for (int j = 0; j < i; j++) {  
 // Intercambiar los elementos de la fila i con los elementos de la columna j  
  
 // Guardar temporalmente el valor de la posición i, j en la variable aux  
 aux = matriz[i][j];  
  
 // Asignar el valor de la posición j, i al elemento en la posición i, j  
 matriz[i][j] = matriz[j][i];  
  
 // Asignar el valor guardado en aux al elemento en la posición j, i  
 matriz[j][i] = aux;  
 }  
 }  
  
  
 // Imprimir la matriz transpuesta  
 System.*out*.println("Matriz transpuesta");  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 System.*out*.print(matriz[i][j] + "\t");  
 }  
 System.*out*.println();  
 }  
 }  
}



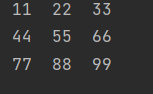
# 124. Creando la matriz transpuesta de una matriz

public class EjemploMatricesTransponerMatriz {  
 public static void main(String[] args) {  
  
 int[][] a, b;  
 a = new int[8][4]; // Declarar matriz a con dimensiones 8x4  
 b = new int[4][8]; // Declarar matriz b con dimensiones 4x8  
  
 // Llenar la matriz a e imprimir sus elementos  
 for(int i = 0; i < a.length; i++){  
 for(int j = 0; j < a[i].length; j++){  
 a[i][j] = i+j\*3; // Asignar valor a cada elemento de la matriz a  
 System.*out*.print(a[i][j] + "\t"); // Imprimir el valor del elemento seguido de una tabulación  
 }  
 System.*out*.println(); // Imprimir una nueva línea después de cada fila  
 }  
  
 // Transponer la matriz a y guardarla en la matriz b  
 for(int i = 0; i < a.length; i++){  
 for(int j = 0; j < a[i].length; j++){  
 b[j][i] = a[i][j]; // Intercambiar elementos entre la fila i y la columna j  
 }  
 }  
  
 // Imprimir la matriz transpuesta b  
 System.*out*.println("La matriz transpuesta b:");  
 for(int i = 0; i < b.length; i++){  
 for(int j = 0; j < b[i].length; j++){  
 System.*out*.print(b[i][j] + "\t"); // Imprimir el valor del elemento seguido de una tabulación  
 }  
 System.*out*.println(); // Imprimir una nueva línea después de cada fila  
 }  
 }  
}



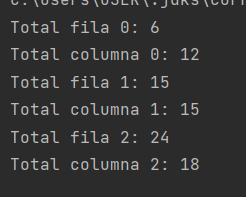
# 125. Operaciones aritméticas en matrices

public class EjemploMatricesSumar {  
 public static void main(String[] args) {  
 int[][] a = {  
 {1,2,3},  
 {4,5,6},  
 {7,8,9}  
 };  
 int[][] b = {  
 {10,20,30},  
 {40,50,60},  
 {70,80,90}  
 };  
 int[][] suma = new int[3][3]; // Declarar matriz suma con dimensiones 3x3  
  
 // Sumar los elementos correspondientes de las matrices a y b y guardar el resultado en la matriz suma  
 for(int i = 0; i < a.length; i++){  
 for(int j = 0; j < a[i].length; j++){  
 suma[i][j] = a[i][j] + b[i][j]; // Sumar elementos correspondientes de a y b y guardar el resultado en suma  
 }  
 }  
  
 // Imprimir la matriz suma  
 for(int i = 0; i < suma.length; i++){  
 for(int j = 0; j < suma[i].length; j++){  
 System.*out*.print(suma[i][j] + "\t"); // Imprimir el valor del elemento seguido de una tabulación  
 }  
 System.*out*.println(); // Imprimir una nueva línea después de cada fila  
 }  
 }  
}



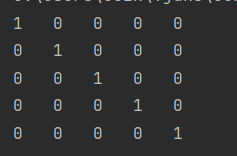
# 126. Sumar filas y columnas de una matriz

public class EjemploMatricesSumarFilasColumnas {  
 public static void main(String[] args) {  
 int sumaFila, sumaColumna; // Variables para almacenar la suma de cada fila y columna  
 int[][] a = {  
 {1,2,3},  
 {4,5,6},  
 {7,8,9}  
 };  
  
 for(int i = 0; i < a.length; i++){  
 sumaColumna = 0; // Reiniciar la suma de la columna en cada iteración del bucle externo  
 sumaFila = 0; // Reiniciar la suma de la fila en cada iteración del bucle externo  
 for(int j = 0; j < a[i].length; j++){  
 sumaFila += a[i][j]; // Sumar los elementos de la fila i y guardar el resultado en sumaFila  
 sumaColumna += a[j][i]; // Sumar los elementos de la columna i y guardar el resultado en sumaColumna  
 }  
 System.*out*.println("Total fila " + i + ": " + sumaFila); // Imprimir la suma de la fila i  
 System.*out*.println("Total columna " + i + ": " + sumaColumna); // Imprimir la suma de la columna i  
 }  
  
 }  
}

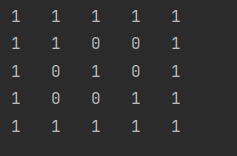


# 127. Límites inferior y superior de la matriz y la diagonal principal

public class EjemploMatricesIdentidad {  
 public static void main(String[] args) {  
  
 int[][] matriz = new int[5][5]; // Crear una matriz de tamaño 5x5  
  
 // Llenar la matriz con elementos de la identidad  
 for(int i = 0; i < matriz.length;i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 if(i == j){  
 matriz[i][j] = 1; // Asignar 1 en la posición (i, j) si i es igual a j  
 }  
 }  
 }  
  
 // Imprimir la matriz  
 for(int i = 0; i < matriz.length; i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 System.*out*.print(matriz[i][j] + "\t"); // Imprimir el elemento en la posición (i, j) seguido de una tabulación  
 }  
 System.*out*.println(); // Imprimir una nueva línea al final de cada fila  
 }  
 }  
}



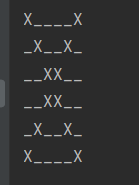
public class EjemploMatricesMarco {  
 public static void main(String[] args) {  
  
 int[][] matriz = new int[5][5]; // Crear una matriz de tamaño 5x5  
  
 // Llenar la matriz con un marco y diagonal de 1  
 for(int i = 0; i < matriz.length; i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 if(i == 0 || i == matriz.length - 1  
 || j == 0 || j == matriz[i].length -1 || i == j){  
 matriz[i][j] = 1; // Asignar 1 en las posiciones de los bordes y la diagonal principal  
 }  
 }  
 }  
  
 // Imprimir la matriz  
 for(int i = 0; i < matriz.length; i++){  
 for(int j = 0; j < matriz[i].length; j++){  
 System.*out*.print(matriz[i][j] + "\t"); // Imprimir el elemento en la posición (i, j) seguido de una tabulación  
 }  
 System.*out*.println(); // Imprimir una nueva línea al final de cada fila  
 }  
 }  
}



# 128. Descargar Código Fuente

# Tarea 18: Imprimir una X

//Escribir un programa que imprima una X construida a base de la letra X en las diagonales de la matriz y utilizar el carácter guion bajo como relleno. El tamaño de la x se basa en una variable n que indicará el tamaño de la letra para imprimir en una matriz de n x n.  
//  
// El carácter "X" en mayúscula y el guion bajo "\_" para los espacios.  
//  
// Por ejemplo para n=5 se obtiene:  
//  
// X\_\_\_X  
// \_X\_X\_  
// \_\_X\_\_  
// \_X\_X\_  
// X\_\_\_X  
// Por ejemplo para n=6 se obtiene:  
//  
// X\_\_\_\_X  
// \_X\_\_X\_  
// \_\_XX\_\_  
// \_\_XX\_\_  
// \_X\_\_X\_  
// X\_\_\_\_X  
// Si n es igual a cero imprimir "ERROR" y finalizar la ejecución del programa.  
  
public class ProgramaLetraX {  
 public static void main(String[] args) {  
  
 int n = 6; // Tamaño de la letra X  
  
 if (n == 0) {  
 System.*out*.println("ERROR");  
 return; // Finalizar la ejecución del programa si n es igual a cero  
 }  
  
 char[][] matriz = new char[n][n]; // Crear una matriz de tamaño n x n  
  
 // Llenar la matriz con guiones bajos  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 matriz[i][j] = '\_'; // Asignar guión bajo en todas las posiciones de la matriz  
 }  
 }  
  
 // Colocar la letra X en las diagonales de la matriz  
 for (int i = 0; i < matriz.length; i++) {  
 matriz[i][i] = 'X'; // Colocar X en la diagonal principal  
 matriz[i][matriz.length - 1 - i] = 'X'; // Colocar X en la diagonal secundaria  
 }  
  
 // Imprimir la matriz  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 System.*out*.print(matriz[i][j]); // Imprimir el elemento en la posición (i, j)  
 }  
 System.*out*.println(); // Imprimir una nueva línea al final de cada fila  
 }  
 }  
}



# Tarea 19: Imprimir un silla

public class ProgramaSilla {  
 public static void main(String[] args) {  
  
 int n = 10; // Tamaño de la silla  
  
 if (n == 0) {  
 System.*out*.println("ERROR");  
 return; // Finalizar la ejecución del programa si n es igual a cero  
 }  
  
 int[][] matriz = new int[n][n]; // Crear una matriz de tamaño n x n  
  
 // Llenar la matriz con ceros  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 matriz[i][j] = 0; // Asignar cero en todas las posiciones de la matriz  
 }  
 }  
  
 // Construir la silla en la matriz  
 for (int i = 0; i < matriz.length; i++) {  
 matriz[i][0] = 1; // Rellenar la primera columna con unos  
 matriz[matriz.length / 2][i] = 1; // Rellenar la fila central con unos  
 matriz[i][matriz.length - 1] = 1; // Rellenar la última columna con unos  
 }  
  
 // Imprimir la matriz  
 for (int i = 0; i < matriz.length; i++) {  
 for (int j = 0; j < matriz[i].length; j++) {  
 System.*out*.print(matriz[i][j]); // Imprimir el elemento en la posición (i, j)  
 }  
 System.*out*.println(); // Imprimir una nueva línea al final de cada fila  
 }  
 }  
}

