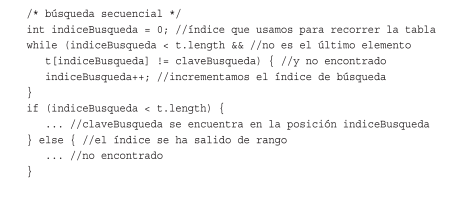
**TEORÍA**

**Muestra el contenido del Array.**

* **Arrays.toString(nombreVariable);**

**NEW.** Añadir una variable

**NULL.** Anula los valores (“vacío”).

**BÚSQUEDA EN TABLA NO ORDENADA (P.139)**

**FILL:** Pone el mismo valor en todo el Array

* **Arrays.fill();**

**SORT (FUNCIONES INTERESANTES):** Ordena en Array (P.138)

* **Arrays.sort();**

**binarySearch** (BÚSQUEDA EN TABLA ORDENADA)

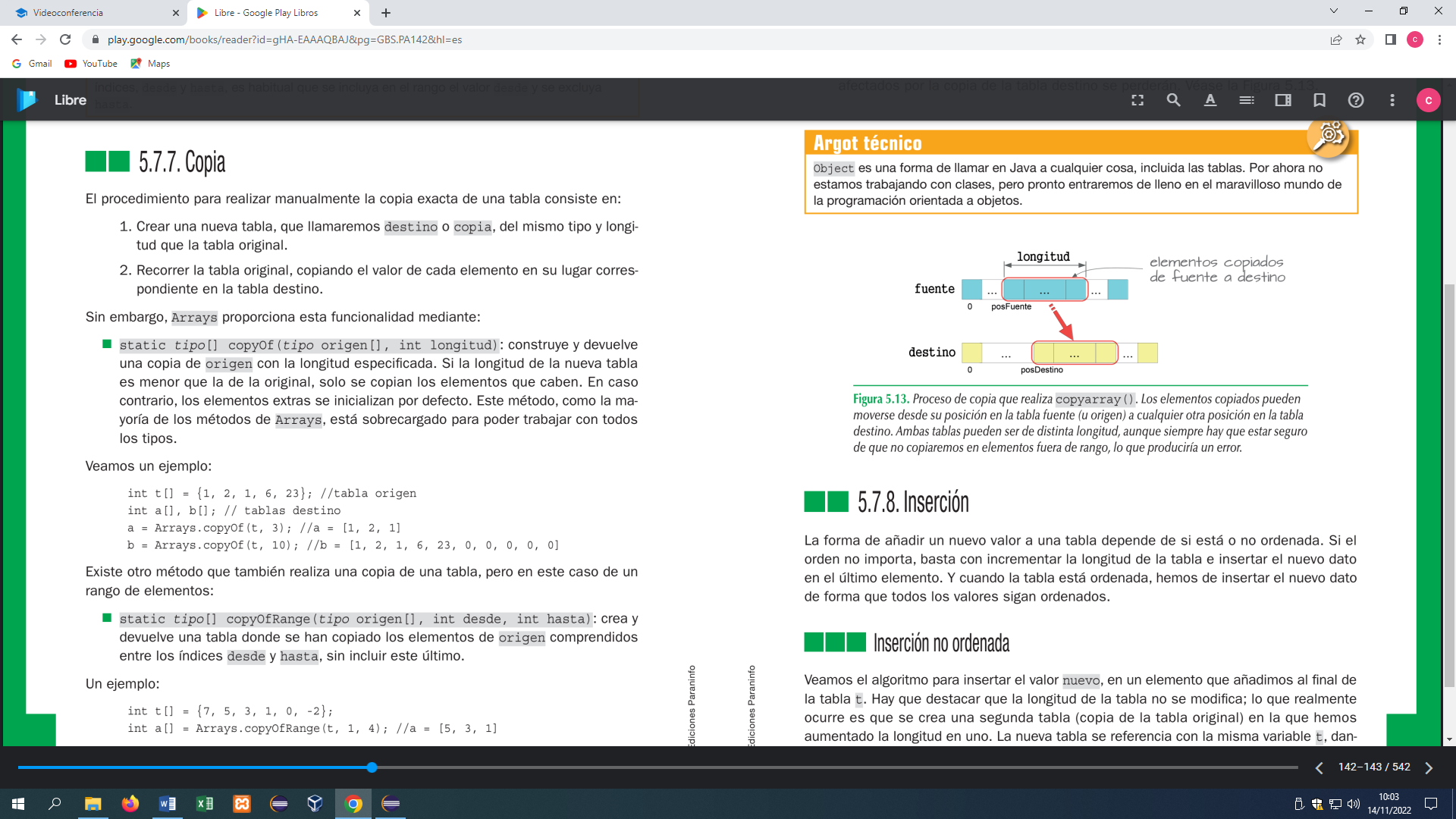
Te dice dónde lo encuentra y si está o no en el Array, si da negativo es que no está en el Array. En caso de que apareciera, te coloca la posición en la que aparecería, para que estuvieran los valores ordenados.

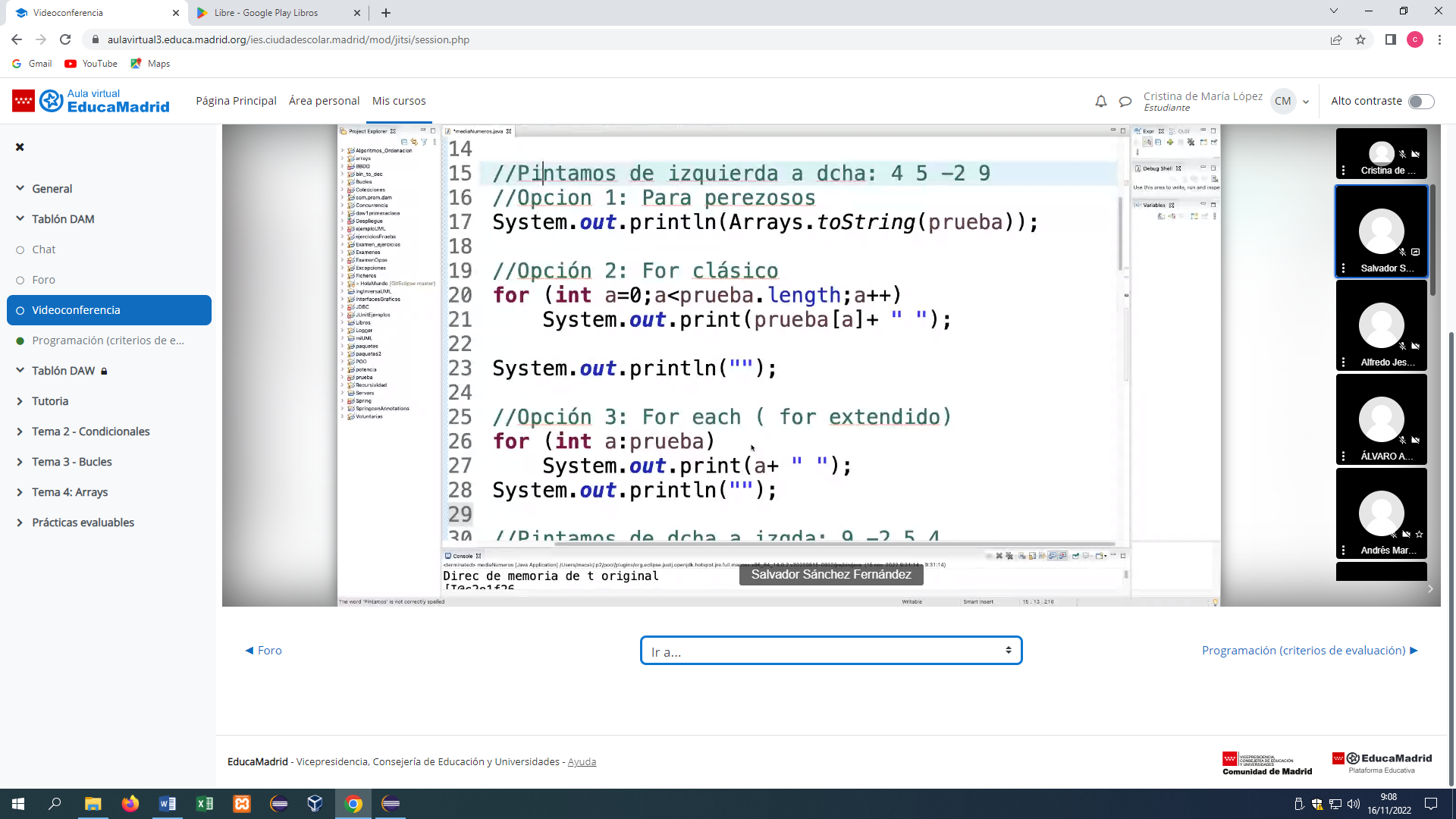
**COPIOF (INTERESANTE) P.142 =**

* **copiOf (origen[], int longuitud)**

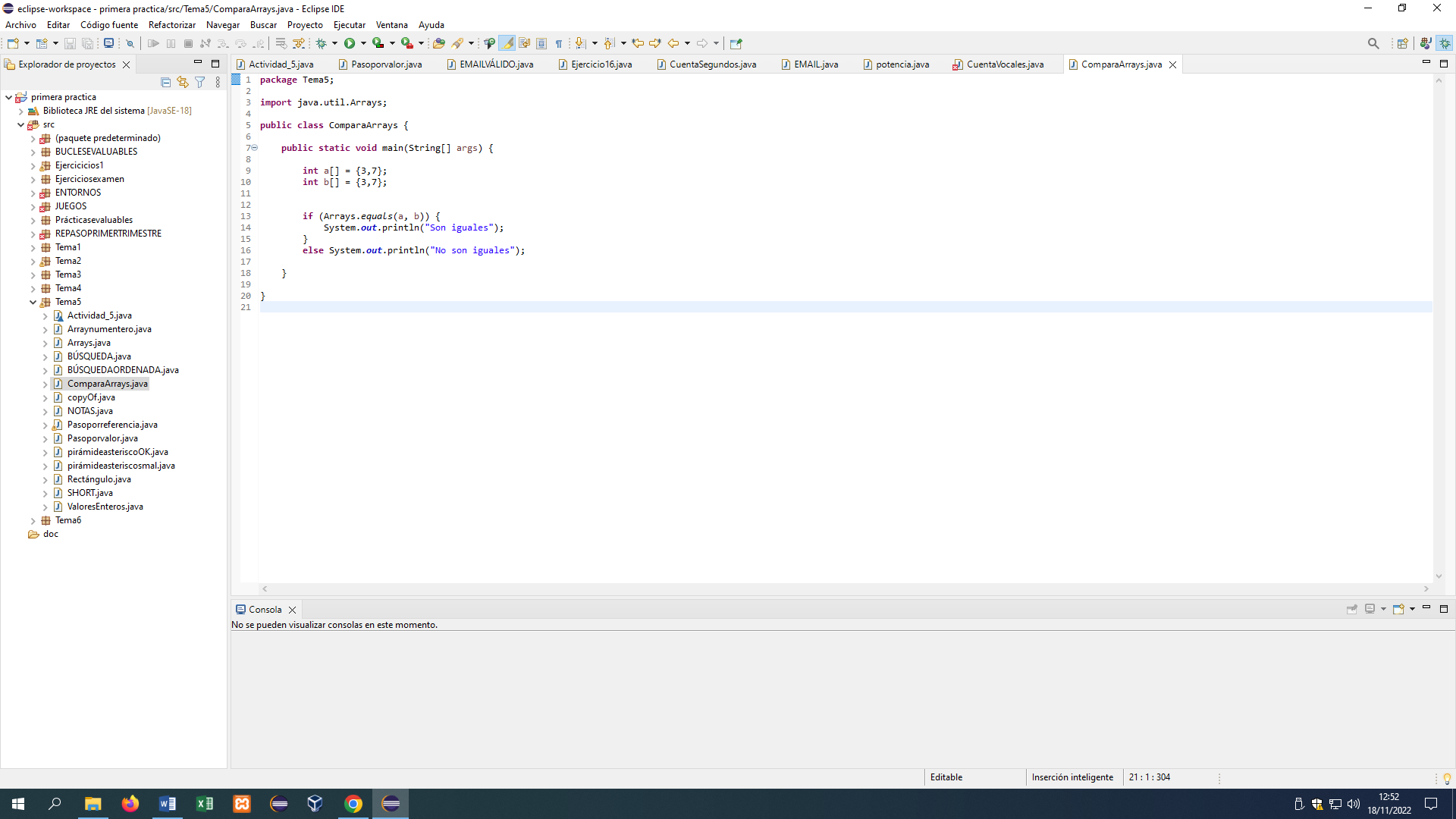
Devuelve la primera ocurrencia.

Copiar Arrays (para hacer trampa)



**FOR EACH**

Recorrido estructuras, no se le da ni el inicio ni el fin, tiene dos partes en el for, a la derecha es el array que recorres (derecha de los puntos), y a la izquierda el mismo tipo de array (Ej. Int)



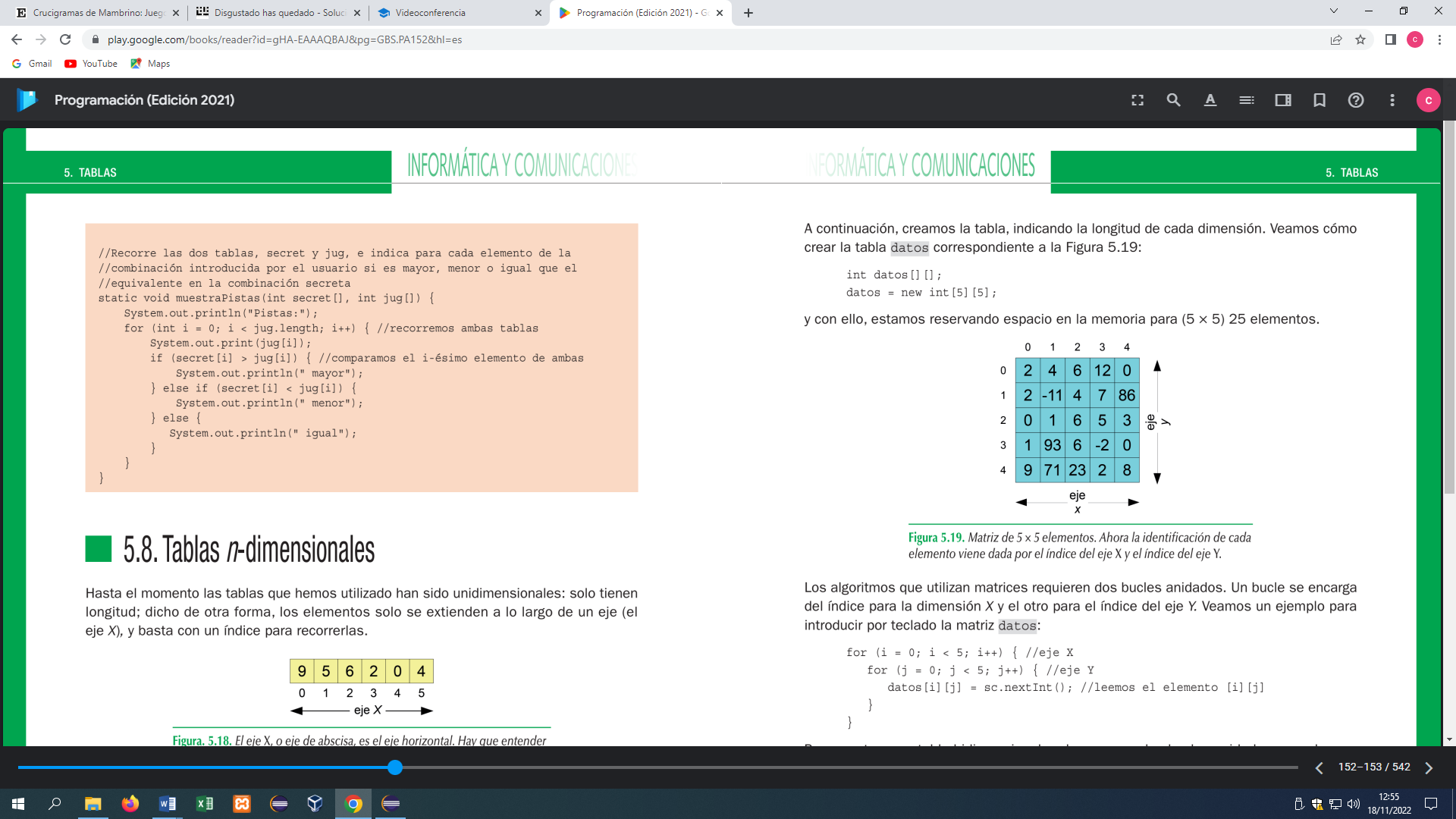
**18/11/2022 COMPARACIÓN DE DOS TABLAS**

**Equals =** comparar String, devuelve un boolean, cuando ambos arrays son iguales, comparamos con dos arrays enteros (t1, t2)

**Static boolean equals (tipo a[], tipo b[])**

**TABLAS DOS DIMENSIONES P.150**

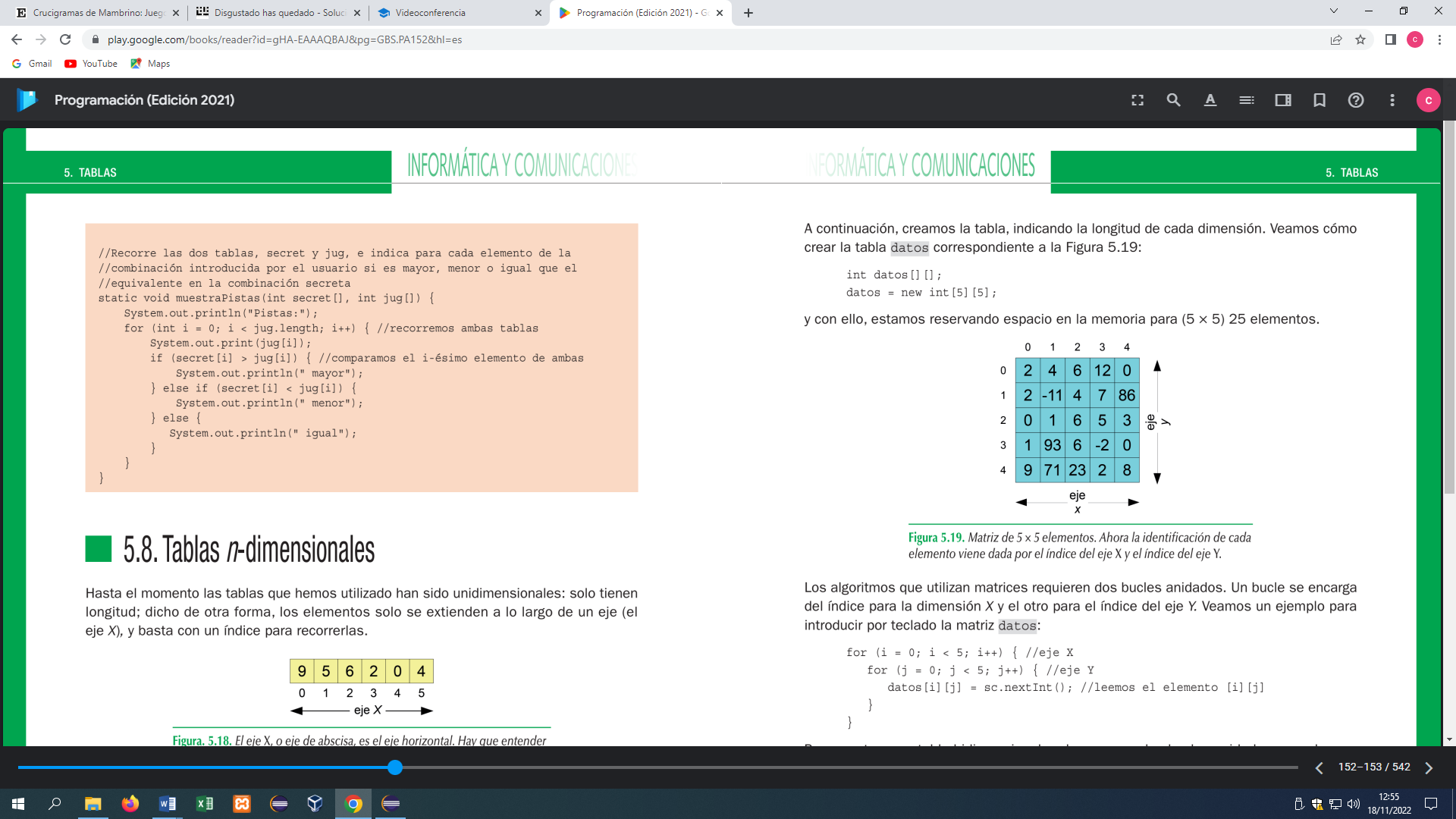
Pones dos corchetes [], []. Primero filas y luego columnas.

Columnas

Filas

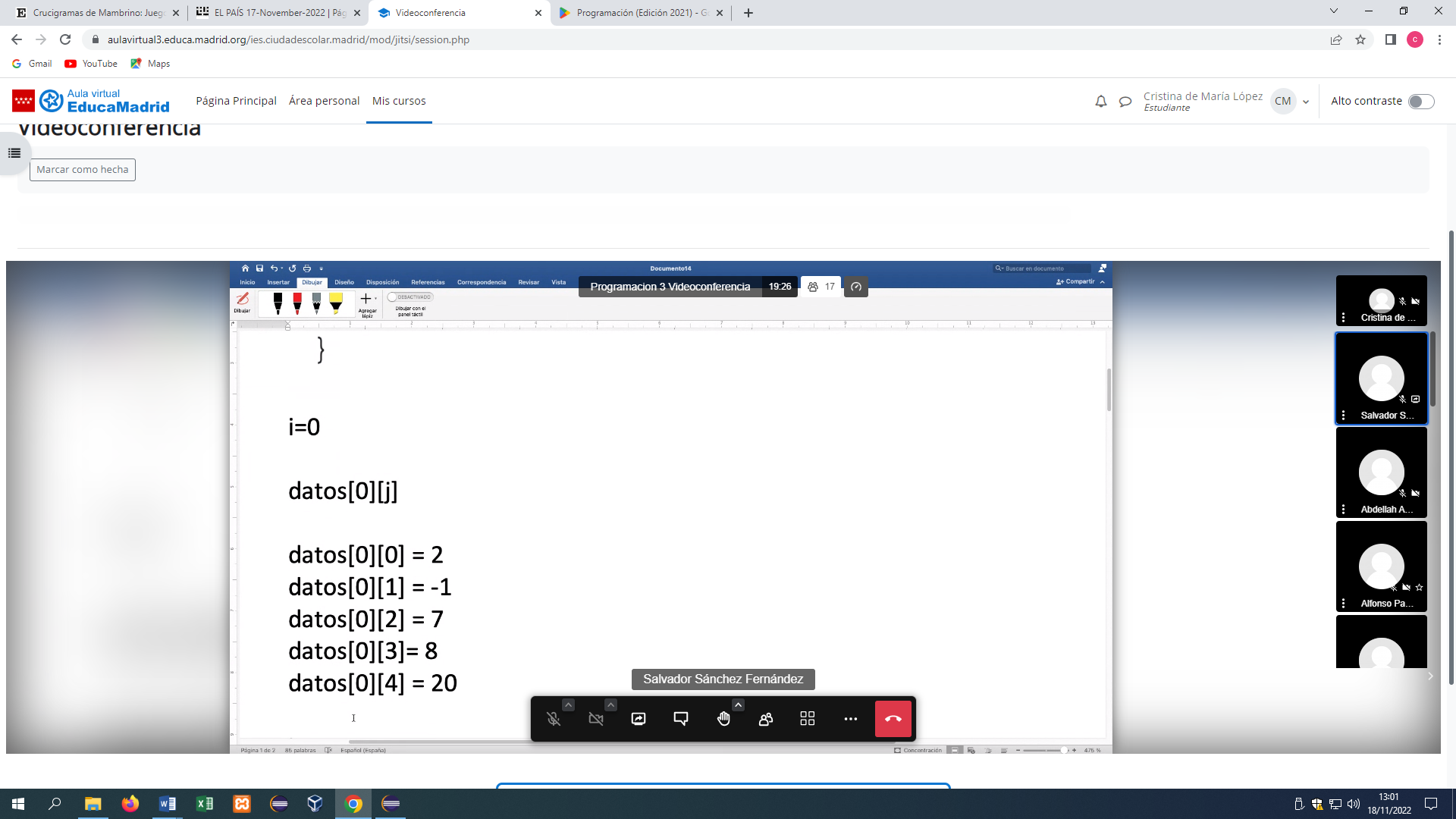
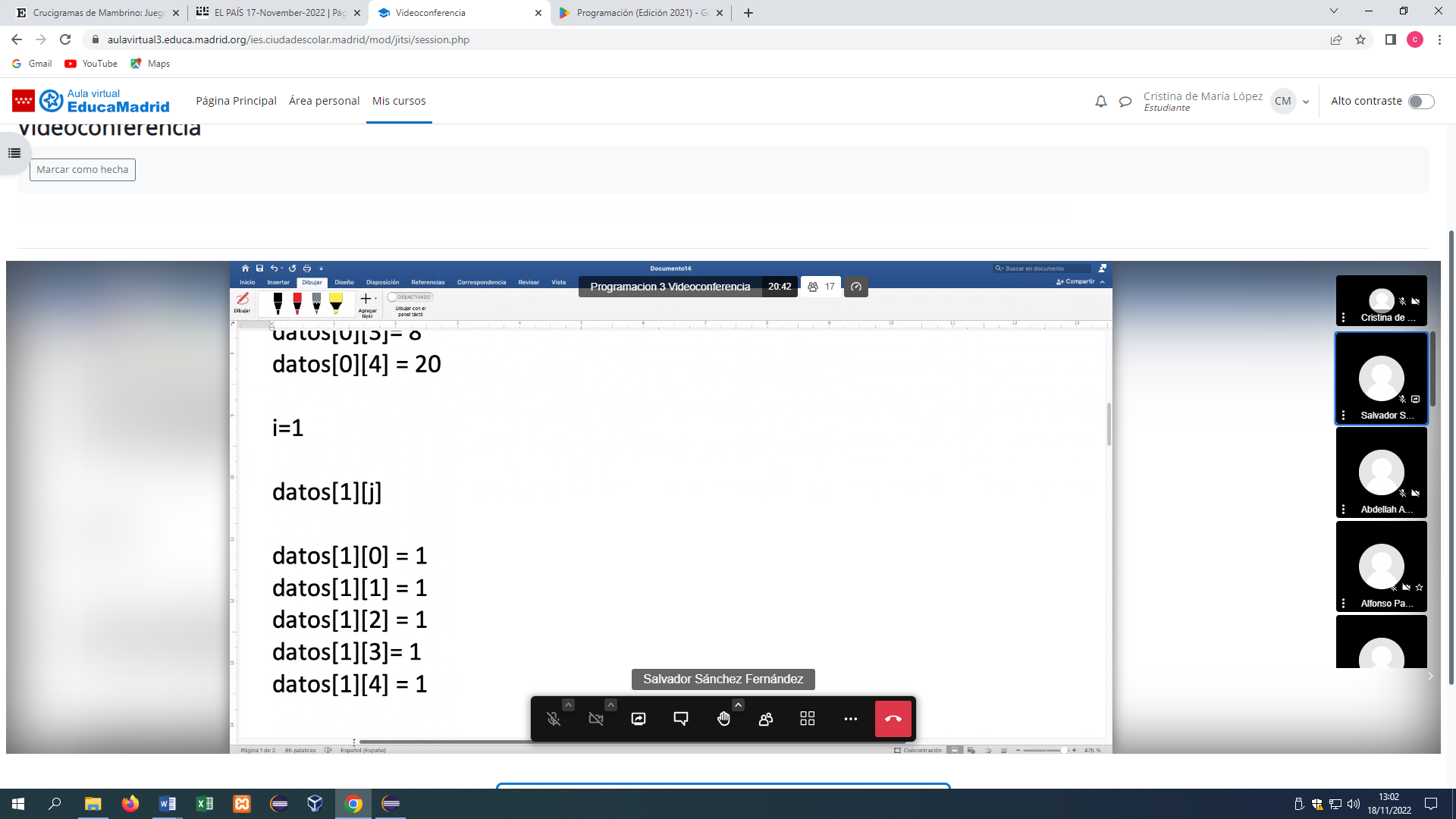
Ej.

Datos [2] [1] = 7;

Y = columnas

X = filas

Utiliza bucles anidados

Y = 0, se queda estático

2 -1 7 8 20

1 1 1 1 1

**COMO SE PINTA UN ARRAYS**

Dos for con la “i” y “j”, y añades un “if”, y haces un Syso, con los valores que se coge, mientras una de las variables se queda estática.

**SEGUNDA FORMA DE HACERSE**

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

**CON UN FOR EACH**

