

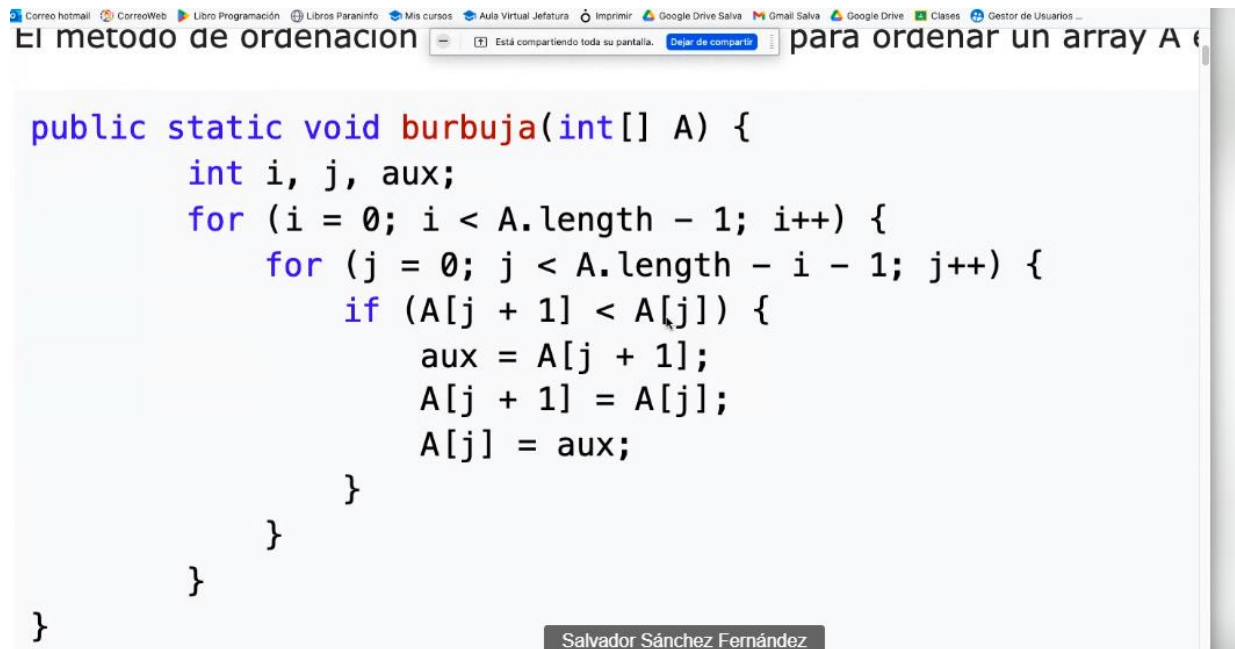
TEMA 5. ALGORITMO BURBUJA

01/12/2022

Revisa cada elemento de la lista que va a ser ordenada.

Están descolocados, los revisa y los va colocando de menor a mayor, los va comparando y se intercambian.

- Intercambio directo.
- Revisa varias veces la lista.



```
public static void burbuja(int[] A) {  
    int i, j, aux;  
    for (i = 0; i < A.length - 1; i++) {  
        for (j = 0; j < A.length - i - 1; j++) {  
            if (A[j + 1] < A[j]) {  
                aux = A[j + 1];  
                A[j + 1] = A[j];  
                A[j] = aux;  
            }  
        }  
    }  
}
```

Salvador Sánchez Fernández

ORDENACIÓN POR SELECCIÓN

Array desordenado, recorre el array con bucle anidado, no llega al final.

Int a[50,27,8,2]

```
//método java de ordenación por selección
public static void seleccion(int A[]) {
    int i, j, menor, pos, tmp;
    for (i = 0; i < A.length - 1; i++) {           // tomamos como menor el primer
        menor = A[i];                               // de los elementos que quedan
        pos = i;                                    // y guardamos su posición
        for (j = i + 1; j < A.length; j++){         // buscamos en el resto
            if (A[j] < menor) {                       // del array algún elemento
                menor = A[j];                         // menor que el actual
                pos = j;
            }
        }
        if (pos != i){                               // si hay alguno menor se intercambia
            tmp = A[i];
            A[i] = A[pos];
            A[pos] = tmp;
        }
    }
}
```

Salvador Sánchez Fernández

INSERCIÓN

1 dimensión, segundo valor y vas comparando con el anterior, si este es menos que el anterior se intercambia.

QUICK SORT

Se elige un numero, y tardará mas o menos en función de cual eligas. (pivot)

```
{
    // TODO Apéndice de método generado automáticamente
    int numeros[] = {1, 9, 23, 4, 55, 100, 2, 1, 24};
    System.out.println("Antes de QS: " + Arrays.toString(numeros));
    quicksort(numeros, 0, numeros.length - 1);
    System.out.println("Después de QS: " + Arrays.toString(numeros));
}

public static void quicksort(int A[], int izq, int der) {
    int pivote=A[izq]; // tomamos primer elemento como pivote
    int i=izq;         // i realiza la búsqueda de izquierda a derecha
    int j=der;         // j realiza la búsqueda de derecha a izquierda
    int aux;

    while(i < j){
        while(A[i] <= pivote && i < j) i++; // busca elemento mayor que pivote
        while(A[j] > pivote) j--;           // busca elemento menor que pivote
        if (i < j) {                        // si no se han cruzado
            aux= A[i];                      // los intercambia
            A[i]=A[j];
            A[j]=aux;
        }
    }
    // System.out.println(Arrays.toString(A));

    A[izq]=A[j]; // se coloca el pivote en su lugar de forma que tendremos
    A[j]=pivote; // los menores a su izquierda y los mayores a su derecha

    if(izq < j-1)
        quicksort(A, izq, j-1); // ordenamos subarray izquierdo
    if(j+1 < der)
        quicksort(A, j+1, der); // ordenamos subarray derecho
}
}
```

Salvador Sánchez Fernández