

TALLER PROBLEMAS DE BÚSQUEDA Y ORDENAMIENTO

SORT THE PEOPLE

DIEGO ARMANDO PEDROZO VILLARREAL

1151898

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

FACULTAD DE INGENIERÍA

PROGRAMA DE INGENIERÍA DE SISTEMAS

ANÁLISIS DE ALGORITMOS

SAN JOSÉ DE CÚCUTA

2022-2

Taller Problemas Búsqueda y Ordenamiento

- 1) Resolver el problema en la plataforma <https://leetcode.com/>
- 2) Desarrollar una aplicación que genere al menos 100 casos de prueba para el problema. Los casos deben cubrir todas las posibilidades de casos del problema, de manera equilibrada.
- 3) Desarrollar una aplicación completa, con main y lectura de datos, de manera que se pruebe la solución independientemente de la plataforma <https://leetcode.com/>. Esta solución debe ser consistente con la solución realizada en el numeral 1.
- 4) Elaborar un video explicando en detalle los tres numerales anteriores. Preferiblemente subirlo a Youtube.
- 5) Publicar en github todo lo anterior y escribir un informe en PDF que se sube a UVIRTUAL con los links a todo lo anterior.

Problema en LeetCode

Enlace → <https://leetcode.com/problems/sort-the-people/>

Se le proporciona un arreglo de nombres y un arreglo de alturas que consta de enteros positivos distintos. Ambas arreglos son de longitud n .

Para cada índice i , `nombres[i]` y `alturas[i]` denotan el nombre y la altura de la i -ésima persona.

Devuelve los nombres ordenados en orden descendente por la altura de las personas.

Planteamiento para la solución del problema

Para la solución del ejercicio se planteó utilizar hashmap porque se especificaba en el ejercicio que cada estatura era única en el array, el uso de hashmap permitía poder enlazar cada nombre de una persona con su correspondiente estatura utilizando esta como identificador único.

Para decidir el orden de manera descendente del nuevo array de nombres solicitado se decidió utilizar el algoritmo de ordenamiento burbuja modificándolo para que ordenará de mayor a menor el array de estaturas.

Luego de obtener el array de estaturas ordenado simplemente se iban insertando los nombres en el nuevo array ordenado de nombres dependiendo de su identificador estatura asignado por medio de hashmap.

Solución LeetCode

```
class Solution {
    public String[] sortPeople(String[] names, int[] heights) {

        Map<Integer, String> personas = new HashMap();
        for(int i = 0; i < heights.length; i++){
            personas.put(heights[i], names[i]);
        }
        burbuja(heights);

        String [] nombres = new String[names.length];
        for(int i = 0; i < heights.length; i++){
            nombres[i] = personas.get(heights[i]);
        }
        return nombres;
    }

    public 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;
                }
            }
        }
    }
}
```

Time Submitted	Status	Runtime	Memory	Language
10/28/2022 12:38	Accepted	40 ms	54.8 MB	java

Submission Detail

68 / 68 test cases passed.

Runtime: 40 ms

Memory Usage: 54.8 MB

Status: Accepted

Submitted: 2 days, 4 hours ago

Solución Github

Enlace → <https://github.com/Diego-Pedrozo/Sort-The-People>

Método	Función
<pre>public static String[] sortPeople(String[] names, int[] heights) { Map<Integer, String> personas = new HashMap(); for (int i = 0; i < heights.length; i++) { personas.put(heights[i], names[i]); } burbuja(heights); String[] nombres = new String[names.length]; for (int i = 0; i < heights.length; i++) { nombres[i] = personas.get(heights[i]); } return nombres; }</pre>	Método utilizado para enlazar los nombres y las estaturas de cada persona por medio de hashmap para luego devolver un array de nombres dependiendo del array de estaturas ya ordenado de manera descendente.
<pre>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; } } } }</pre>	Método burbuja convencional con una mínima modificación para ordenar el array de estaturas de manera descendente.
<pre>public static String[] generarNombres(int cantidad) { String[] nombresAleatorios = new String[cantidad]; String[] nombres = { "Andrea", "David", "Baldomero", "Balduino", "Baldwin", "Baltasar", "Barry", "Bartolo", "Bartolome", "Baruc", "Baruj", "Candelaria", "Candida", "Canela", "Caridad", "Carina", "Carisa", "Caritina", "Carlota", "Baltazar" }; String[] apellidos = { "Gomez", "Guerrero", "Cardenas", "Cardiel", "Cardona", "Cardoso", "Cariaga", "Carillo", "Carion", "Castiyo", "Castorena", "Castro", "Grande", "Grangenal", "Grano", "Grasia", "Griego", "Grigalva" }; for (int i = 0; i < cantidad; i++) { nombresAleatorios[i] = nombres[(int) (Math.floor(Math.random() * ((nombres.length - 1) - 0 + 1) + 0))] + " " + apellidos[(int) (Math.floor(Math.random() * ((apellidos.length - 1) - 0 + 1) + 0))]; } return nombresAleatorios; }</pre>	Método utilizado para generar el array de nombres aleatorios.

```
public static int[] generarEstaturas(int cantidad) {
    int[] estaturas = new int[cantidad];
    for (int i = 0; i < cantidad; i++) {
        estaturas[i] = (int) (Math.random() * 200) + 1;
    }
    return estaturas;
}
```

Método utilizado para generar el array de estaturas aleatorias.

Caso Borde

Un caso borde identificado con el uso de hashmap es cuando más de una persona tiene la misma estatura, esto se debe a que el hashmap asigna a cada nombre su estatura correspondiente como identificador único y cuando se requiere asignar a otro nombre una estatura ya en uso se reemplaza el nombre anterior con el nuevo.

```
Digite el tamaño del arreglo
4
Digite el array de nombres
Andres Kenn Diego Joan
Digite el array de estaturas
165 175 170 170
Andres,165
Kenn,175
Diego,170
Joan,170

Personas Ordenadas por estatura(Descendentemente)
Kenn
Joan
Joan
Andres
```

Todos los Enlaces

1. Ejercicio en LeetCode → <https://leetcode.com/problems/sort-the-people/>
2. Github → <https://github.com/Diego-Pedrozo/Sort-The-People>
3. Video explicativo → <https://youtu.be/BLX2Y8hlc7I>