



Lugar y números

Para practicar la implementación del algoritmo de búsqueda binaria, vamos a empezar con un ejemplo simple. Queremos que se familiaricen con la sintaxis del mismo, por lo que no haremos énfasis en los datos, sino en la solución.

Dada la siguiente lista:

```
let list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11];
```

Utilizar el algoritmo de búsqueda binaria para responder los siguientes ítems:

- ¿Cuál es la posición del número 1?
- ¿Cuál es la posición del número 5?
- ¿Cuál es la posición del número 6?
- ¿Cuál es la posición del número 9?
- ¿Cuál es la posición del número 11?

Sabemos que responder estas preguntas es extremadamente fácil, ya que podemos leer el array, y determinar con un cálculo visual la posición de cada elemento, pero, la propuesta que les hacemos es que codeen un algoritmo de búsqueda binaria, que “busque” ese número, por ejemplo, el 6, y nos indique por consola la posición del mismo.



El objetivo de este ejercicio es que puedan practicar la sintaxis sin añadir complejidad extra.

Desafío extra: Orden, lugar y números

Aumentemos la dificultad un toque, y apliquemos lo aprendido en semanas anteriores. Queremos que hagan lo mismo del ejercicio anterior —buscar la posición de un número en un array—, pero partiendo de esta lista:

```
let list = [12,3,5,7,1,22,47,100];
```

Para aplicar búsqueda binaria, deberán ordenar primero la lista, de menor a mayor, utilizando bubble sort. Luego, respondan las siguientes preguntas:

- ¿Cuál es la posición del número 12?
- ¿Cuál es la posición del número 5?
- ¿Cuál es la posición del número 22?
- ¿Cuál es la posición del número 100?

Podrán realizar este ejercicio en un Code Sandbox o Node. Escoger el ambiente en el que se sientan con más comodidad.