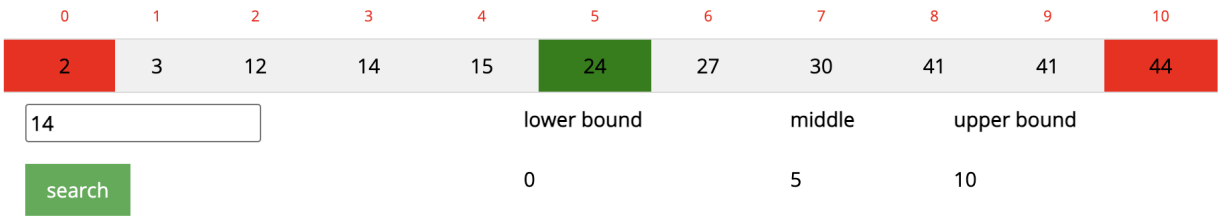
# Binary Search

## Scenario

An algorithm needs to search an array for an item, 14.




Binary search is often used to search **linear data structures** such as 1 dimensional arrays.

*Note: Binary search only works on **sorted** data sets.*

It is a “divide-and-conquer” algorithm.

It starts by determining 3 important items of data:

- the *lower bound* (smallest index of the array)
- the *upper bound* (largest index of the array)
- the *middle index* (the index of the item in the middle of the array

```java
int[] nums = {2,3,12, 14, 15, 24, 27, 30, 41, 41, 44};

int lb = 0;
int ub = nums.length-1;
int midIndex = nums.length/2;
```

While the item at the middle index is not the same as the search item, the algorithm divides the array as follows:

*If the item being searched for is greater than the middle item, the lower bound is updated to **midIndex+1.***
*If the item being searched for is less than the middle item, the lower bound is updated to **midIndex-1**.*

*The middle index is then **recalculated**.*

```java
int[] nums = {2,3,12, 14, 15, 24, 27, 30, 41, 41, 44};

int searchItem = 14;

int lb = 0;
int ub = nums.length-1;
int midIndex = (lb+ub)/2;

while(nums[midIndex] != searchItem){
	if(searchItem > nums[midIndex]){
		lb = midIndex + 1;
	}

	if(searchItem < nums[midIndex]){
		ub = midIndex - 1;
	}

	//recalculate midIndex
	midIndex = (lb+ub)/2;
}

//end of loop, display midIndex and or item at midIndex
System.out.println(searchItem+" found at index "+midIndex);
```

**Task 1**
Open your idea and run the above code. Does it successfully find an item in the array?

**Task 2**
Mess around with the values in the array so that the array is not sorted. Does the algorithm still operate successfully?

**Task 3**

What if the search item is not in the **sorted array**? Does the algorithm still work? If not, can you fix the problem so that a message is displayed, "*Item not found in the array*".

**Task 4**

A method `getStringIndex` returns the index of a String in an array of Strings. If the search item is not in the array, the method returns -1.

*The array may be initially unsorted.*

Complete the following program in your IDE:

```java
public static void main(String[] args){
	String[] words = {}; //you can populate the array with random words

	sortWords(words);

	int result = getStringIndex(words, "hello");

	System.out.println("Index: "+result);
}

public static void sortWords(String[] wordList){
	//use a sorting algorithm to sort the array
}

public static int getStringIndex(String[] wordList, String searchItem){

	//implement this method
}
```

**Task 5**

What is the Big O of Binary Search in terms of Time and Space?

**Time:**

**Space:**