**Exponential search:**

public class Searching {

boolean exponentialSearch(int arr[], int key) {

int lengthOfArray = arr.length;

if (arr[0] == key) { // Checking whether first element is the key

return true;

}

// Finding the range in which the element might be present

int i = 1;

while (i < lengthOfArray && arr[i] <= key) {

i = i \* 2; // Exponentially increasing value of i.

}

return binarySearch(arr, i / 2, Math.min(i, lengthOfArray - 1), key); // calling binary search method on the sub-array

}

boolean binarySearch(int arr[], int low, int high, int key) {

int mid; // to store middle element

while (low <= high) {

mid = (low + high) / 2; // we can also do mid = low+(high-low)/2 to avoid overflow in some cases

if (arr[mid] == key) {

return true;

} else if (arr[mid] < key) {

low = mid + 1;

} else {

high = mid - 1;

}

}

return false;

}

// Driver Code

public static void main(String args[]) {

Searching search = new Searching();

int arr[] = {

1,

3,

4,

6,

8,

13,

15,

24

};

if (search.exponentialSearch(arr, 15)) {

System.out.println("Element found !");

} else {

System.out.println("Element not found :( ");

}

}

}

**Output:**

Element found !