

```

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Objects;

public class BinarySearch{
    // 複数の Integer 型を引数として受け取れる（可変長引数で調べてみると良いでしょう）。
    // 引数で受け取った複数の Integer 型の変数を格納した ArrayList を作成して返す。
    ArrayList<Integer> create(Integer... values){
        Arrays.sort(values); // ソートする。
        ArrayList<Integer> list = new ArrayList<>();
        for(Integer value: values){
            list.add(value);
        }
        return list;
    }

    // list から value を探す。見つければ、そのインデックスを返す。
    // 見つからない場合、-1 を返す。
    void demo(ArrayList<Integer> list, Integer value){
        Integer index = search(list, value, 0, list.size() - 1);
        System.out.printf("search(%, %d): %d%n", list, value, index);
    }

    void runDemo() {
        demo(create(1, 2, 3, 5, 8, 13, 21), 5); // (1), (2)
        demo(create(1, 3, 5, 7, 11, 13, 15), 13); // (3), (4)
        demo(create(1, 3, 7, 11, 13, 17, 19), 23); // (5), (6)
    }

    // 二分探索。
    Integer search(ArrayList<Integer> list, Integer value,
        Integer low, Integer high) {
        if(low <= high) {
            Integer middle = (low + high) / 2;
            Integer middleValue = list.get(middle);
            if(Objects.equals(value, middleValue)){
                return middle;
            }
            if(value < middleValue) {
                return search(list, value, low, middle - 1);
            }
            return search(list, value, middle + 1, high);
        }
        return -1;
    }

    public static void main(String[] args){
        BinarySearch bs = new BinarySearch();
        bs.runDemo();
    }
}

```