COP 3503 Recitation #2

Problem

Given a sorted array of n distinct values as well as a target value T, determine in *O(n) time* whether or not there exist two distinct values in the array that sum to T. (For example, if the array contained 3, 5, 6, 7, and 9 and T = 14, then the method you are to write should return true, since 5+9 = 14. It should return false if for the same array of values T = 17.)

Input File Format

The first line of the file will have a single positive integer k, representing the number of test cases in the file. The next 2k lines will contain the test cases, with two lines being used for each test case. The first value on the first line of each test case will be n, the size of the array. The rest of the line will contain n distinct integers sorted in ascending order, each separated by spaces. The second line of each test case will contain a single integer, T, the target for the problem. Here's an example:

```
2
5 3 5 6 7 9
14
3 1 3 6
11
```

Output Format

For each test case, output a line with one of the following two formats:

```
Test case #m: The target T is achievable.
Test case #m: The target T is NOT achievable.
```

where m ($1 \leq m \leq k$), represents the appropriate test case in the file.

For the example above, the output would be:

```
Test case #1: The target 14 is achievable.
Test case #2: The target 11 is NOT achievable.
```

What to Turn In

Your group of two should submit one file named *array.java* which solves the problem specified above. In particular, your program should read its input from a file named *array.in* (which has the file format specified above), and prints the corresponding output to the screen as specified above. Please provide ample comments in your code, as well as an argument as to why it runs in O(n) time, where n represents the number of values in the array. If you don't find such an algorithm, solve the problem another way for partial credit, and do the analysis for that solution instead. (Namely, don't try to argue that a solution that isn't O(n) is O(n), especially if you know that to be the case!!!)