

Task 1.

a.

Use nested for-loops to loop through every possible combination and checking if $a_i + a_j$ would be equal to k . Excluding the combination when $a_i = a_j$ of course.

b.

Sort the array using mergesort and loop (while $I < J$) I and j points to the beginning and end of the array and If $a_i + a_j$ is LESS than K , increment i . if $a_i + a_j$ is GREATER than K , decrement j . if $a_i + a_j$ is EQUAL to K you have found a match and have to increment I and decrement J .

c.

Loop through array and check if $(K - a_i)$ is in a hashmap. If not, then add a_i to hashmap. If it is, then you have a match.

Ps:

- In task 3 I added a “public Pos parent;” in the Pos.java class to make it easy to backtrack to start of path. I spoke with Fredrik Manne and he said it was fine.
- I could also have started from the end and searched for the start. This would eliminate the need for a stack for reversing the result.