

**CS60020: Foundations of Algorithm Design and Machine Learning**  
**Assignment 1**  
**Submission deadline 21<sup>st</sup> Jan 2018**

1. Describe a  $O(n \log n)$  - time algorithm that, given a set  $S$  of  $n$  integers and another integer  $x$ , determines whether or not there exist two elements in  $S$  whose sum is exactly  $x$ .

2. Given an array  $A[1..n]$  containing  $n$  distinct integers. If  $i < j$  and  $A[i] > A[j]$ , then  $(i,j)$  called an inversion in  $A$ . Write a **divide and conquer algorithm** to count number of inversions in  $A$ .

**Example:**

For array  $A=[2,4,1,3,5]$ , the number of inversions are 3.

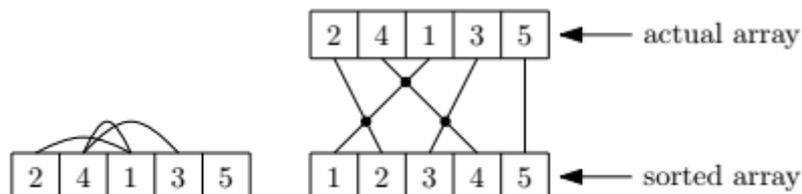


Figure 1.1: Inversion  $\iff$  crossing pair.

3. Solve the asymptotic upper bound for the given equation:

$$T(n) = \begin{cases} 4T(\frac{n}{2}) + 7n & \text{if } n > 4 \\ 12 & \text{otherwise.} \end{cases}$$