Algorithms (COT 6405): Assignment 2

Due date: September 4 (Thursday)

Problem 1 (5 points)
Let $A[1..n]$ be a *sorted* array of n distinct numbers. Write an efficient algorithm Binary-Search(A, n, k) that finds a given value k in $A[1..n]$. It should return the index of the found element; for example, if $A = \langle 1, 3, 4, 6, 9 \rangle$ and $k = 6$, then the returned index is 4, which means that $k = A[4]$. If the array does not include k, the algorithm should return 0.

Problem 2 (5 points)
Let $A[1..n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an *inversion*. For example, the array $\langle 2, 3, 8, 6, 1 \rangle$ contains five inversions. Write an algorithm Inversions(A, n) that determines the number of inversions in $A[1..n]$.