CS 4349 ASSIGNMENT I

1. Write a proof by induction to show the correctness of the binary search code given below:

```
// Find index of x in sorted array A[p..r]. Return -1 if x is not
in A[p..r].
int binarySearch ( A, p, r, x ):  // Pre: A[p..r] is sorted
  if p > r then return -1
  else
      q <-- (p+r)/2
  if x < A[q] then
        return binarySearch ( A, p, q-1, x)
      else if x = A[q] then
        return q
      else // x > A[q]
        return binarySearch ( A, q+1, r, x)
```

[10 Points]

2. Consider sorting n numbers stored in array A by first finding the smallest element of A and exchanging it with the element A[1]. Then find the second smallest element of A, and exchange it with A[2]. Continue in this manner for the first n-1 elements of A. Write pseudocode for this algorithm, which is known as selection sort. What loop invariant does this algorithm maintain? Why does it need to run only the first n-1 elements, rather than for all n-elements? Give the best-case and worst-case running times of selection sort in Θ-notation. **[10 Points]**

3. Write a pseudocode describing a Θ(n lg 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. **[10 Points]**

4. For the following two problems use induction to prove. Recall the standard definition of the Fibonacci numbers:

$$F_0 = 0, F_1 = 1 \; and \; F_n = F_{n-1} + F_{n-2} \; for \; all \; n \geq 2.$$

a. Prove that $\sum_{i=0}^{n} F_i = F_{n+2} - 1$ for every non-negative integer n. **[10 Points]**

b. The Fibonacci sequence can be extended backward to negative indices by rearranging the defining recurrence: $F_n = F_{n+2} - F_{n+1}$. Here are the first several negative-index Fibonacci numbers:

n	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1
F_n	-55	34	-21	13	-8	5	-3	2	-1	1

Prove that $F_{-n} = (-1)^{n+1} F_n$ for every non-negative integer n. **[10 Points]**