1. (10%) Design an algorithm to rearrange elements of a given array of n real numbers so that all its negative elements precede all its positive elements. Your algorithm should be both time- and space-efficient.

2. (10%) Write the postfix and prefix forms of the following expression: $(a+b)*d+e/(f+a*d)+c$.

3. (15%) (a) Construct a Huffman code for the following data:

character	A	B.	C	D	—
probability	0.4	0.1	0.2	0.15	0.15

   (b) Encode the text ABACABAD using the code of question (a).
   (c) Decode the text whose encoding is 100010111001010 in the code of question (a).

4. (10%) For the following list, construct an AVL tree by inserting their elements successively, starting with the empty tree: 1, 2, 3, 4, 5, 6.

5. (15%) For the input 30, 20, 56, 75, 31, 19 and hash function $h(K) = K \bmod 11$.

   (a) Construct the open addressing hash table using linear probing.
   (b) Find the largest number of key comparisons in a successful search in this table.
   (c) Find the average number of key comparisons in a successful search in this table.

6. (15%) Apply the branch-and-bound algorithm to solve the traveling salesman problem for the following graph.
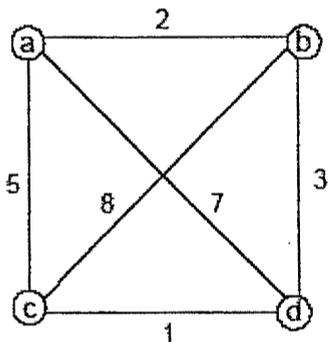
7. (15%) Consider the following recursive algorithm.

   Algorithm Secret($A[0..n-1]$)
   　if $n = 1$ return $A[0]$
   　else $temp \leftarrow$ Secret($A[0..n-2]$)
   　　if $temp \leq A[n-1]$ return $temp$
   　　else return $A[n-1]$

   (a) What does this algorithm compute?
   (b) Set up a recursive relation for the algorithm's basic operation count and solve it.

8. (10%) Sort the following list by heapsort by using the array representation of heaps: S, O, R, T, I, N, G (in alphabetical order).