INSTRUCTIONS
Students should only discuss this with the instructional staff and their homework group.
KEY CONCEPTS Sorting, Asymptotic notation, loop invariants.

1. Review problem for homework 1: Give a closed form expression for $\sum_{i=1}^{i=n} 3i^2 + 3i + 1$ (5 points) and prove your answer by induction (10 points). Then use this to give a closed form expression for $\sum_{i=1}^{i=n} i^2$ (5 points)

2. For each of the following claims, say whether it is true or false, and give a short explanation (2 points correct answer, 2 points explanation).
   (a) $2^n \in O(4^n)$
   (b) $n^2 \in o(n^4)$
   (c) $2^{n/2} \in O(2^n)$
   (d) $(n^3 + n + 2)^3 \in \Theta(n^8)$.
   (e) $\sum_{i=1}^{\lceil \log_2 n \rceil} 2^i \in \Theta(n)$.

3. Consider the problem of given two sorted lists of distinct integers, $A[1] < . < A[n]$ and $B[1] < .. < B[n]$, return the number of elements that are in both lists. Design an algorithm that solves the problem. (10 points) Use a loop invariant to prove that your algorithms is correct. (10 points). Give a time analysis in O notation for your algorithm (10 points). If the lists were given to you unsorted, what would be an efficient method to solve this problem and how long would it take? (10 points).

4. For each algorithm, compute the runtime in terms of Θ. . (show your work., 10 points each)
   (a)
   ```
   for i = 1 to n:
       j = 1
       while j < n do:
           j = 2 * j
           print (i, j)
   ```
   (b)
   ```
   i = 0
   j = 0
   while  i < n:
       print i.
       i = i + 3j^2 + 3j + 1
       j = j + 1
   ```

5. Design a "binary search style" algorithm for the following problem: The input is an array of integers $A[1..n]$ so that $A[1] \neq A[n]$. The problem is to find an integer I so that $1 \leq I \leq n-1$ and $A[I] \neq A[I+1]$. Use loop invariants to show that your algorithm correctly solves this problem, and that it takes $O(logn)$ time.