1. Every permutation of $n$ elements determines a set of inversions. Prove the converse: each permutation is uniquely determined by its set of inversions. **Hint:** Try to prove by contradiction. In other words – assume this is not the case and prove that it cannot be true.

2. Decide whether each of the following statements is true or false, and prove that your conclusion is correct.
   (a) $2^{n+1} = O(2^n)$
   (b) $f(n) = O(g(n))$ implies $2^f(n) = O(2^g(n))$

3. Prove that $\log_a x = O(\log_b x)$ for any $a > 0$ and $b > 0$.

4. Prove that if $f = O(g)$ and $g = O(h)$ then $f = O(h)$.

5. Problem 4-1 (a, b, c, f, g) (page 107 in 3rd edition, 119 in 4th edition).

6. Problem 4.2 in Lecture notes 1 (page 7).

7. Problem 4.1 in Lecture notes 2 (page 12).

8. (a) Show how to implement a stack (Last in first out) using a Priority Queue.
   (b) Show how to implement a queue (First in first out) using a Priority Queue.

   In both cases, do not worry about an $O(1)$ run time.

9. Describe an $O(n \log k)$ algorithm for merging $k$ sorted lists into one sorted list, where $n$ is the total number of elements. **Hint:** Think of the merging part of MergeSort and extend it to multiple lists. Remember that the lists are not necessarily the same size.

10. The Split3-Sort algorithm is defined as follows:

```
Algorithm 1 Split3-Sort(A,p,r)
1: if (A[p] > A[r]) then
3: end if
4: if (p + 1 > r) then
5:   k = \lfloor (r - p + 1)/3 \rfloor // Round down
6:   Split3 - sort(A, p, r - k) // First two thirds
7:   Split3 - sort(A, p + k, r) // Last two thirds
8:   Split3 - sort(A, p, r - k) // First two thirds again
9: end if
```
(a) Prove that the call to Split3-Sort(A, 1, n) correctly sorts the array A[1..n] (Hint: I found the best way is to use induction, but be careful with the base case - notice line 4. What is the minimum difference between p and r?)

(b) Write the recurrence formula for Split3-Sort and give the asymptotic bound on the run time (∈ notation).

(c) Compare the run time from (b) to the run time of HeapSort, MergeSort and QuickSort. Is it better? Worse? Same?