CPSC 320: Tutorial 6

1. Suppose that we are given an array A of n integers, and an integer k. Describe an algorithm that will find the k elements of A closest numerically (not by their order statistics) to the median of A. The running time of your algorithm should be $O(n)$.

   Hint: How can you write, mathematically, that an element x is closer numerically to the median m than another element y?

2. The algorithm **GoodPivot** divides its n element input array A into $n/5$ groups of 5 elements each (and maybe one group with less than 5 elements). This allowed us to prove that there were at least $\lfloor 3n/10 \rfloor$ input elements less than the pivot. We then used this fact to derive a recurrence relation that upper bounds the running time of the algorithm **BSelect** and established a $O(n)$ upper bound on the solution of that recurrence.

   Suppose that we had decided to divide the input elements into $n/3$ groups of 3 elements each instead. How many element are guaranteed to be less than the pivot? Justify your answer.

   Derive a recurrence relation that upper bounds the running time of this new version of **BSelect**. What is the best upper bound you can get for the solution of this recurrence?

   If you have time, do the same for groups of 7 elements.