Quiz 1

Suppose you are given an array $A[1..n]$ of $n$ distinct integers and some fixed $k$, $1 \leq k \leq n$. You should think of $k$ as small and $n$ as arbitrarily large. We seek the $k$ largest elements of $A$. For example, if $n=5$ and $k=2$ and $A$ is the array:

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>12</td>
<td>75</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

then the answer would be 75 and 22.

**a** Use the decision tree model (the basic operation is pairwise comparisons) to give a lower bound on the worst-case complexity of the problem.

**b** Give a linear (in $n$) time upper bound on the worst-case complexity of this problem.

**Solutions:**

**a** Each subset of $k$ elements of $A$ is a possible answer, so the decision tree must have at least $\binom{n}{k}$ leaves. Fixing $k$, we note that $\binom{n}{k} = \frac{n(n-1)...(n-k+1)}{k!} \in \Omega\left(n^k\right)$. So the height of the tree must be in $\Omega\left(\lg n^k\right) = \Omega\left(k \lg n\right)$, which is $\Omega\left(\lg n\right)$ for fixed $k$. Note that the bound is not tight.

**b** We find $A$'s largest element, and return it and remove it, $k$ times.

```
for i ← 1 to k do
    MAXINDEX ← 1

for j ← 2 to n-1 do

return A[MAXINDEX]

A[MAXINDEX] ← −∞
```