Question 1.

Using the KMP algorithm, show how to search for pattern $P =$ acbcacb in text $T =$ aabacbcacbacbcacbcac: Using the indices of P and T list which of characters of P are compared with which characters of T.

Question 2.

a) Consider the following optimization problem. Describe the problem as a decision problem.

Longest Path
Input: An edge-weighted graph $G = (V, E)$
Output: A simple path of maximum length (measured by the sum of the weights of its edges) in G

b) Consider the following decision problem. Describe the problem as an optimization problem.

Clique
Input: A graph $G = (V, E)$ and an integer k
Question: Does there exist a clique $V' \subseteq V$ in G of size at least k? Here, a subset $V' \subseteq V$ of vertices is a *clique* for G if for each pair of vertices $x, y \in V'$: $(x,y) \in E$.

Question 3.

As discussed in class, a subset V' of the vertices V, $V' \subseteq V$, of a graph $G = (V, E)$ is a vertex cover for G if for every edge (x,y) in the graph: $x \in V'$ or $y \in V'$. This means that for every edge, at least one of its at points is in the vertex cover.

Further, we discussed the following observation that yields a branching rule.

Observation 1. Let V' be a vertex cover for graph $G = (V, E)$, and let $x \in V'$. Then either $x \in V'$, or $x \notin V'$. In the latter case, if $x \notin V'$, then all of the vertices adjacent to x, that is the set $\mathrm{N}(x)$, are in V': $\mathrm{N}(x) \subseteq V'$.

From this we derived a recursive algorithm, based on Observation 1 and the fact that vertices of degree zero do not have to be included in the cover, using a decision tree (also called search tree) that is of size $\mathrm{O}(2^k)$.

Now consider the following observation (also discussed in class):

Observation 2. Let V' a vertex cover for graph $G = (V, E)$, and let $(x,y) \in E$. Then $x \in V'$, or $y \in V'$.

a) Describe a search-tree algorithm based solely on Observation 2.
b) What is the size of your algorithm's search tree? Argue convincingly.