Advanced Algorithms (TMS)

(a) Given an algorithm for the SET-COVER problem as a black box, how could you use this to solve the unweighted VERTEX-COVER problem? [4 marks]

(b) Following the approach in Part (a), which approximation ratio for the VERTEX-COVER problem do you achieve by applying the greedy algorithm for the SET-COVER problem? What happens if every vertex in the graph has at most 4 neighbours? [6 marks]

(c) Consider the following greedy algorithm for the unweighted VERTEX-COVER problem:

Compute a directed Depth-First-Search tree (DFS-tree) from every connected component in the graph, and output all nodes which are not leaves in the DFS-tree (a vertex is a leaf if it has no outgoing edges in the DFS-tree).

(i) What is the running time of this algorithm? [2 marks]

(ii) Why is the returned solution a valid vertex cover? [4 marks]

(iii) Derive a bound, as good as possible, on the approximation ratio of this algorithm.

Hint: You may use the fact that in any undirected graph $G = (V,E)$, $\sum_{u \in V} \deg(u) = 2|E|$, where $\deg(u)$ denotes the number of neighbours of $u$. [4 marks]