Algorithms

Final Exam: 7 June 2022

1. A *palindrome* is a nonempty string that reads the same forward and backward. Examples of palindromes are all strings of length 1, `civic`, and `racecar`. Give an efficient algorithm to find a longest palindrome that is a subsequence of a given input string. For example, given the input `character`, your algorithm should return `carac`. What is the time complexity of your algorithm?

2. Write the definition of d_{ij}^k and the recurrence for d_{ij}^k in the Floyd-Warshall algorithm. How can the output of the Floyd-Warshall algorithm be used to detect the presence of a negative-weight cycle?

3. Show how the algorithm Strongly-Connected-Components works on the following graph G. Show the discover/finish times computed in DFS(G). Assume that DFS considers vertices in alphabetical order and that the adjacency lists are in alphabetical order.

4. Draw the string-matching automaton for the pattern `abaababb` over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

5. Write the definition of NP-completeness (i.e., a problem L is NP-complete if ...). Show that if an NP-complete problem A is polynomial-time solvable, then $P = NP$.

6. Given that CLIQUE is NP-hard, prove that SUBGRAPH ISOMORPHISM is NP-hard.

   - Problem: SUBGRAPH ISOMORPHISM
   - Input: Two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$
   - Question: Does G contain a subgraph isomorphic to H? (Two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there exists a one-to-one function $f : V_1 \to V_2$ such that $(u, v) \in E_1 \Leftrightarrow (f(u), f(v)) \in E_2$.)