CS3383, Winter 2019 Assignment # 4

Rasoul Shahsavarifar

Faculty of Computer Science, UNB

Due time: Friday, Feb/8/2019, 1:30 p.m

Student's full name: ..... Student ID:.....

Note:

- No submission after the due time will be accepted.
- The full credit will be given only for correct solutions that are described clearly.

Question 1 (6 marks) (Exercise 9.1 of DPU textbook) In the backtracking algorithm for SAT, suppose that we always choose a subproblem (CNF formula) that has a clause that is as small as possible; and we expand it along a variable that appears in this small clause. Show that if the input formula only contains clauses with two literals[1], then a satisfying assignment, if one exist will be find in polynomial time.

Question 2 (6 marks) Design and write, in pseudocode, a backtracking algorithm that solves the following problem (3-D Matching).
Input: A base set X and a set of triples $S \subseteq \{(x, y, z)|x, y, z \in X\}$.
Output: Is there some $S' \subseteq S$ such that every element of X occurs in exactly one triple of S'?

Question 3 (7 marks) Design and write, in pseudocode, a backtracking algorithm that, given a graph $G = (V, E)$ (V : vertices, E : edges) and an integer k, determines if there is a set of k vertices of G such that no two vertices in this set have an edge between them.

Question 4 (6 marks) A Sudoku of rank n is an $n^2 \times n^2$ square grid, subdivided into n^2 blocks, each of these blocks has the size of $(n \times n)$. The problem is that given distinct integer numbers $1, 2, 3, ..., n^2$, fill the grid in with these numbers such that every integer should appear only once in each raw, in each column, and in each block. The Sudoku problem is known to be an NP-complete.

a) Suppose that you are given an instance $(n^2 \times n^2)$ grid filled in with distinct integer numbers $1, 2, 3, ..., n^2$. What is the formula that can be applied to check if the desired conditions are met by the given instance?

b) Design an algorithm to check if the given grid is a solution for the Sudoku problem.

c) Analyze you algorithm in (b).

(Hint: Read the Note 6 on $N-$Queen verification.)

[1] It is an instance of 2SAT.