EECS 3101 Fall 2021 (B & E) – Assignment 4

Remember to write your **full name** and **student number** prominently on your submission. To avoid suspicions of plagiarism: at the beginning of your submission, **clearly state any resources (people, print, electronic) outside of the course materials, and the course staff, that you consulted**. You must submit a PDF file with the required filename. The PDF file could be generated using LaTeX(recommended), Microsoft Word (saved as PDF, **not** the `.docx` file), or other editor/tools. The submission must be **typed**. Handwritten submissions are **not** accepted.

Due December 3, 2021, 10:00 PM (on eClass); required file(s): a4sol.pdf

Answer each question completely, always justifying your claims and reasoning. Your solution will be graded not only on correctness, but also on clarity. Answers that are technically correct that are hard to understand will not receive full marks. Mark values for each question are contained in the [square brackets]. Your submitted file does **not** need to include/repeat the questions — just write your solutions.

The assignment must be completed individually.

1. **[10] Paradox**

   In this question, we will design an algorithm to solve a problem in philosophy.

   A **paradox** is a group of statements that lead to a contradiction. For example, the following two statements form a famous paradox:

   ```
   1. Statement 2 is FALSE.
   2. Statement 1 is TRUE.
   ```

   If we assume Statement 1 is `TRUE`, then Statement 2 is `FALSE`, which in turn means Statement 1 is `FALSE`. But if we assume Statement 1 is `FALSE`, then Statement 2 is `TRUE`, which in turn means Statement 1 is `TRUE`. So Statement 1 is `TRUE` iff Statement 1 is `FALSE`, a blatant contradiction.

   Now, suppose you are given a group of N statements, numbered from 0 to $N-1$. Each statement has the form: "`Statement X is TRUE/FALSE`," where `X` is a number between 0 and $N-1$. Your task is to figure out whether this group of statements forms a paradox. In particular, answer the following questions.

   (a) Describe how to construct a graph to solve this problem. Be precise: state clearly what vertices your graph contains (and what each vertex represents) and what edges your graph contains (and what each edge represents).

   (b) Give a **necessary and sufficient** condition for the N statements to form a paradox. Justify your answer.

   (c) How do you efficiently detect whether your graph from Part (a) satisfies your condition described in Part (b)? Describe your algorithm in concise (but precise) English.

   (d) Analyze the worst-case runtime of your algorithm.

   (e) Provide the pseudocode of your algorithm. The signature of the function should be: `bool paradox(String[] A)`, where the input `A` is an array of strings such as `"Statement 2 is FALSE"`. Assume that the statement number is the same as the array index at which the statement is stored (both starting from 0). Make sure your pseudocode is clean and concise so that it describes your algorithm clearly **without** including unnecessary details such as how the string parsing is implemented.

2. **[10] Bucket Flips**

   In this question, you will design the solution to the general form of a well-known brain-teaser.

   You are given two **initially empty** buckets A and B, with capacities of m litres and n litres, respectively. Your goal is to measure exactly k litres of water using these two buckets. Assume that m, n and k are positive **integers** and that $k \leq \max(m, n)$. You want to achieve this goal by performing a sequence of **moves** until one of the buckets has exactly k litres of water. Each move can be one of the following:

- Fill a bucket until it's full.
- Empty a bucket.
- Use the water in one bucket to fill the other until one of the buckets is full or empty.

Your job is to design a function `bucket_flip`(m, n, k), which takes m, n and k as inputs and returns the **sequence of moves** to solve the puzzle. The number of moves must be the **minimum** possible. If it is impossible to measure k litres of water using the two buckets, the function returns `NULL`.

Answer the following questions.

(a) Describe how to construct a graph to solve this problem. Be precise: state clearly what vertices your graph contains (and what each vertex represents) and what edges your graph contains (and what each edge represents).

(b) How does your algorithm work? Give a clear and concise description and justify its correctness.

(c) Analyze the worst-case runtime of your algorithm.

(d) Provide the pseudocode of your algorithm. Again, make your pseudocode clean and concise so that it describes your algorithm clearly **without** including unnecessary details.