Homework #4 (Hand-Written)

Purple Correction Date: 12/14/2022 14:00
Blue Correction Date: 12/6/2022 00:00
Red Correction Date: 12/4/2022 00:00
Due Time: 2022/12/20 14:20
Contact TAs: ada-ta@csie.ntu.edu.tw

Instructions and Announcements

- There are **two hand-written problems**.
- You should upload your answer to **Gradescope** as demonstrated in class. For each sub-problem, please label (on Gradescope) the corresponding pages where your work shows up. NO LATE SUBMISSION IS ALLOWED.
- **Collaboration policy.** Discussions with others are strongly encouraged. However, you should write down your solutions **in your own words**. In addition, for **each and every** problem you have to specify the references (e.g., the Internet URL you consulted with or the people you discussed with) on the first page of your solution to that problem. You may get zero point due to the lack of references.

Problem 1 - Laser Tank (Hand-Written) (30 points)

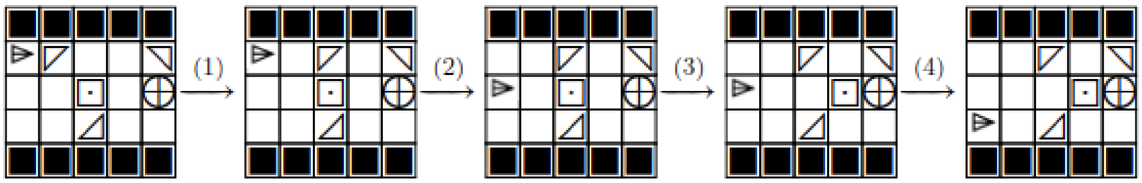
Laser tank is a turn-based puzzle game. You can get a playable version here. Just like many other puzzle games (e.g., tetris and minesweeper), deciding if a laser tank puzzle is solvable is NP-hard. In the following section, you'll reduce a 3-CNF satisfiability problem instance to a puzzle of this game and prove it's NP-hard.

Introducing the game

Laser tank is a turn-based puzzle game played on a 2D grid (the *board*). Each turn, the player can either move the tank or fires a laser. The laser interacts with different *pieces* on the board, and the goal is to hit a certain *piece* with the laser. The *pieces* we use are *mirrors* (the right-angled triangles), *solid blocks* ■, *movable blocks* ⊡, the *tank* ⊳, and the *goal* ⊕.

Tank (⊳) is the only *piece* controlled by the player with two actions: move or fire. In our version, the movement of tank is restricted to vertical axis (i.e., the tank can only move up or move down instead of all four directions). The tank can fire a laser from the front (i.e. to the right). If the laser hits a mirror on a slanted edge, it is reflected. When a mirror is hit on one of the two (non-reflective) short edges by the laser, the mirror is pushed along the direction of the laser. A movable block is pushed one step if it is hit by the laser. A movable block or a mirror is only pushed if the tile directly behind it is empty. The aim of the puzzle is to hit the goal piece with the laser. The solid blocks do not allow lasers or the tank to pass through and they do not move when hit by the laser.

Here is a small instance of the problem, with a step-by-step solution. The tank fires a laser which moves a mirror (1), then takes moves one step down, (2). It then shoots a laser at the movable block (3), and finally moves in position to have a shot of the goal (4). At this moment, emitting a laser solves the problem.



Gadget

A gadget is a small part of the board with some tiles specified to be input tiles and output tiles. For convenient, please follow the below icon convention. A tile is said to be shootable if the tank can shoot the position with movements but without pushing any piece in advance (e.g. in the illustration above, the goal is not shootable before step 1, but is shootable after step 4).

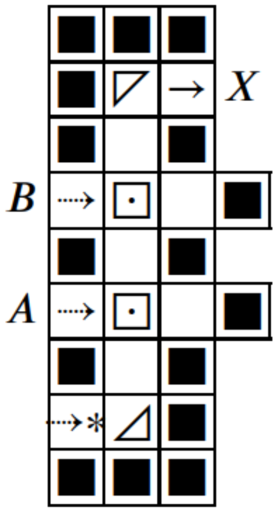
1. Use dotarrows to indicate input.
2. Use dotarrows with star to indicate tiles always shootable.
3. Use arrows to indicate output.

The subproblems

A blank page filled up with grid is provided, you can print it out and draw your answers on it to save time. In the "gadget construction" subproblems, please draw your construction clearly and briefly explain you answer **within 10 lines**.

(0) Example subproblem: AND Gadget (0 point)

Please design a gadget with two input tiles and one output tile such that the output tile can become shootable if and only if both input tiles are shootable. This gadget would serve as the "AND" gate for 3-CNF problem.
Example answer:



(1) OR Gadget (4 points)

Please design a gadget with three input tiles and one output tile such that the output tile can become shootable if and only if at least one among the three input tiles is shootable. This gadget would serve as the "3-way OR" gate.

(2) Literal Gadget (4 points)

Please design a gadget with two output tiles such that

- Both tiles can become shootable from the beginning state.
- Once one output tile becomes shootable, the other tile can no longer be shootable in the future.

This gadget guarantees `(X || !X) == true` and `(X && !X) == false`.

(3) Switch Gadget (2 points)

Please design a gadget with one input tile , one always shootable tile and two output tiles such that

- In the begining: The first output tile is shootable if and only if the input tile is shootable, while the second output tile is not shootable.
- Once after the always shootable tile is shooted: The first tile is not shootable, while the second tile is shootable if and only if input tile is shootable.

This gadget enables you to reuse the "shootability" of an output tile from other gadget.

From here on, please do **NOT** draw the details inside the gadgets. To utilize a gadget, just draw a box and label the name of the gadget and specify I/O tiles. You can cite the above-mentioned gadgets even if you didn't answer those subproblems.

(4) Practicing Laser (4 points)

Construct a game corresponding to the formula `(A || B) && !A && !B`, the shootability of goal should map to the satisfiability of the formula.

Do not evaluate truth value of the formula. Answers just draw a random laser tank puzzle, claiming it has same solvability to the truth value of formula would not be accepted. The answer puzzle should contain structure corresponding to logic literals and connectives of the formula.

P.s. N-way OR (N: positive integer) has basically the same structure, so use 2-way OR without its construction is acceptable in this subproblem.

(5) 3-SAT can be Laser tank puzzle (8 points)

Design an algorithm that runs in polynomial time and reduces an instance of 3-CNF formula's satifiability problem to a laser tank puzzle.

(6) Constructive reduction proof (8 points)

Design two algorithms. The first one converts "a boolean assignment satisfying the 3-CNF formula" to "a solution to the puzzle you constructed in (5)". The second one restores "a satisfying boolean assignment to the original 3-CNF-SAT problem" from "a solution to the laser tank puzzle". With these algorithms, you can conclude that the 3-CNF-SAT problem is reduced to laser tank puzzle, which implies the laser tank puzzle is NP-hard.

Problem 2 - Pledge of Z (Hand-Written) (20 points)

Majin Buu is going crazy and is ready to bombard our Earth. As one of the Z-Fighters, you, Goku, will not surrender to him. Nevertheless, Buu is so strong that you can't beat him even in Super Saiyan 3 mode. The only way to save the world is to collect as much energy as possible. You soon begin charging up your special move "Spirit Bomb", which requires the inhabitants of Earth to raise their hands to share their energy with you.

Piccolo, your smartest comrade, tells you the relations between the N people on the Earth. That is, Piccolo tells you a set of directed edges $U = \{(s_1, d_1), (s_2, d_2), \ldots, (s_M, d_M)\}$, each element (s_i, d_i) denotes that person indexed s_i can reach d_i. Note that $(s, d) \in U$ does not necessarily imply $(d, s) \in U$.

Since Buu is about to destroy the world, you can only use "telepathy" to persuade K people so that the number of people reachable from these K people is maximized. If you chose the K people optimally, let the number of people reachable from these K people be $\aleph_{\text{opt}}$. Your goal is to select K individuals using a greedy algorithm such that the number of people reachable from these K people is no less than $\left(1 - \left(\frac{K-1}{K}\right)^K\right) \cdot \aleph_{\text{opt}}$.

You find that it's harder to solve this problem than to beat Buu. You don't have time to argue with your comrades or complain anonymously on the Internet due to the emergency. So, you decide to focus on some simple subproblems first.

When solving (a), (b), (c) and (d), you should not consider the original problem.

(a) (3 points)

Given finite sets $A_1, \ldots, A_M$ such that $\left|\bigcup_{i=1}^{M} A_i\right| = N$ and a positive integer $K \le M$. Design a greedy algorithm that runs in $O(NMK^2)$ time and outputs K indices $a_1, a_2, \ldots, a_K$ such that

$$\left|\bigcup_{i=1}^{K} A_{a_i}\right| \ge \max_{1 \le b_1 < b_2 < \cdots < b_K \le M} \left|\bigcup_{i=1}^{K} A_{b_i}\right| \cdot \left(1 - \left(\frac{K-1}{K}\right)^K\right).$$

For simplicity, you may use a function $f : (\text{a set of } K \text{ indices}) \to \mathbf{N} \cup \{0\}$ to denote the size of the union of K sets. More formally, $f(\{t_1, t_2, \ldots, t_K\}) = \left|\bigcup_{i=1}^{K} A_{t_i}\right|$. This function runs in $O(NK)$ time. You don't need to give a proof for correctness in this subproblem – give it in subproblem (d).

(b) (3 points)

Given an instance with $M = 5, K = 3, N = \left|\bigcup_{i=1}^{M} A_i\right|$, where

$$A_i = \begin{cases} X_{i-1} & , \forall i = 1, 2, 3 \\ Y_{i-4} & , \forall i = 4, 5 \end{cases}, \text{ where } \begin{cases} X_i = \bigcup_{j=0}^{8} \{(i, j)\} \\ Y_i = \bigcup_{l=0}^{2} \bigcup_{r=0}^{\rho(i)} \{(l, \sigma(i) + r)\}, \text{ where } \begin{cases} \sigma(i) = 3^2 \left(1 - \left(\frac{2}{3}\right)^i\right) \\ \rho(i) = \left(\frac{2}{3}\right)^i \cdot 3^1 - 1 \end{cases} \end{cases}$$

Find a possible greedy selection a_1, a_2, a_3 by applying algorithm constructed from (a) such that $f(\{a_1, a_2, a_3\}) = \max_{1 \le b_1 < b_2 < b_3 \le 5} f(\{b_1, b_2, b_3\}) \cdot \left(1 - \left(\frac{2}{3}\right)^3\right)$.

(c) (3 points)

Given a positive integer K, please construct sets $A_1, A_2, \ldots, A_M$ such that

$$\max_{1 \le b_1 < b_2 < \cdots < b_K \le M} \left| \bigcup_{i=1}^{K} A_{b_i} \right| = \left| \bigcup_{i=1}^{M} A_i \right| = N = K^K.$$

And, it's possible for your algorithm to output $a_1, a_2, \ldots, a_K$ where

$$\left| \bigcup_{i=1}^{K} A_{a_i} \right| = N \cdot \left(1 - \left(\frac{K-1}{K} \right)^K \right).$$

Also, prove your correctness of your construction.

(d) (3 points)

Prove that

$$\left| \bigcup_{i=1}^{K} A_{a_i} \right| \ge \max_{1 \le b_1 < b_2 < \cdots < b_K \le M} \left| \bigcup_{i=1}^{K} A_{b_i} \right| \cdot \left(1 - \left(\frac{K-1}{K} \right)^K \right)$$

holds for the K indices $a_1, a_2, \ldots, a_K$ returned from your algorithm in (a).

(e) (3 points)

Solve the original problem in $O(NMK)$ time and prove your correctness. You may cite any piece of your proof in (d).

(f) (5 points)

Let's consider a variation of the original problem.

For each element e in U, e appears in U' with constant probability p. Let A_i be the set of people reachable from person i only with the relations in U'. Design an algorithm that runs in $O\left(NMK2^M\right)$ time and output K indices such that

$$E\left[\left| \bigcup_{i=1}^{K} A_{a_i} \right| \right] \ge E\left[\aleph_{\text{opt}}\right] \cdot \left(1 - \left(\frac{K-1}{K} \right)^K \right)$$

Also, prove the correctness. Note that you will not know what U' really is. In other words, you can only make your final solution through p and N, U given in the original problem. Also,

$$E\left[\aleph_{\text{opt}}\right] \tag{1}$$

is the optimal expected reachable people if you chose the K people based on N,U,p optimally. To see an example, please refer to NTU Cool Announcement