CS 577: Introduction to Algorithms Fall 2020

Homework 10

Instructor: Dieter van Melkebeek TA: Nicollas Mocelin Sdroievski

This homework covers NP-completeness of Satisfiability and closely related problems. **Problems 1 and 3 must be submitted for grading by 11:59pm on 11/30.** Problem 1 will be discussed in your discussion section on Friday. Please refer to the homework guidelines on Canvas for detailed instructions.

Warm-up problems

1. [Graded, 2 points] A CNF formula is *monotone* if the literals in each clause are either all positive or all negative. For example, if we have

   - $\varphi_1 = (x_1 \vee x_2 \vee x_3 \vee x_4) \wedge (\overline{x_2} \vee \overline{x_3} \vee \overline{x_4}) \wedge (x_1 \vee x_3 \vee x_4)$ and
   - $\varphi_2 = (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3 \vee \overline{x_4}) \wedge (x_1 \vee x_3 \vee x_4)$,

   then φ_1 is monotone but φ_2 is not.

   The problem of MONO-SAT is the restriction of CNF-SAT to monotone formulas: Given a monotone CNF formula, does it have a satisfying assignment? Show that MONO-SAT is NP-hard.

2. Consider *not-all-equal SAT* (NAE-SAT): Given a CNF formula, decide if there exists an assignment such that each clause contains at least one true literal and one false literal. In NAE-k-SAT, each clause has at most k literals.

   For example, consider

   - $\varphi_1 = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_3} \vee x_1)$ and
   - $\varphi_2 = (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3 \vee \overline{x_4}) \wedge (x_1 \vee x_3 \vee x_4)$.

   Both φ_1 and φ_2 have a satisfying assignment, but φ_1 is a NO instance for NAE-SAT since its only satisfying assignment assigns all literals in the clause $(x_1 \vee x_2 \vee x_3)$ to 1. φ_2 is a YES instance for NAE-SAT since we can assign $x_1 = 1$, $x_2 = 1$, $x_3 = 0$ and $x_4 = 1$.

   (a) Show that NAE-4-SAT is NP-hard using a reduction from 3-SAT.

   (b) Show that NAE-3-SAT is NP-hard using a reduction from NAE-4-SAT.

Regular problems

3. [Graded, 3 points] Due to unfortunate planning, two game development conferences are happening simultaneously in Beijing and Chicago. Each conference is showcasing several games, and each game is supposed to be presented in person by one of its developers. Developers often work on multiple games, and may present on any subset of the games they have worked on. However, it is infeasible for a single developer to attend both conferences in person. The conferences' organizers would like to know whether it is possible to assign presentations to developers so that no developer has to present at both conferences.

   Show that the following problem is NP-complete: Given two lists of games (one per conference), and a list of developers for each game, decide whether it is possible for each conference to have every game on its list be presented by one of its developers such that no developer needs to attend both conferences.

The following are examples of instances for the problem and their respective answers.

Example 1. The conference in Beijing is showcasing games G_1, G_2 and G_3, and the conference in Chicago is showcasing games G_1 and G_4. The following is a list of developers for each game:

- G_1: $\{d_1, d_2\}$.
- G_2: $\{d_1, d_3\}$.
- G_3: $\{d_4\}$.
- G_4: $\{d_3, d_4\}$.

In this example, having d_1 present games G_1 and G_2, and d_4 present G_3 in Beijing, while having d_2 present G_1 and d_3 present G_4 in Chicago is a valid solution, since no developer needs to present at both conferences. In this case the answer would be yes. Note that it is not a problem for d_1 to present games G_1 and G_2 in the same conference. Note also that is is allowed for the two conferences to showcase the same game.

Example 2. The conference in Beijing is showcasing games G_1, G_2 and G_3, and the conference in Chicago is showcasing games G_4 and G_5. The following is a list of developers for each game:

- G_1: $\{d_1, d_3\}$.
- G_2: $\{d_2, d_3\}$.
- G_3: $\{d_1, d_2, d_3\}$.
- G_4: $\{d_3\}$.
- G_5: $\{d_1, d_2\}$.

In this example, it does not matter how we pick developers to present games, we always end up with some developer needing to present at both conferences. Note that we must pick d_3 to present G_4. After that, the only choice for G_2 is d_2, who can also present G_3. We are then left with G_1 in Beijing and G_5 in Chicago, but only a single developer to present both. Therefore, the answer is no.

4. Show that the following problem is NP-hard: Given a 3-CNF formula, does there exist an assignment that satisfies exactly one literal of every clause?

5. Some time ago, people from the programming languages group asked about the following problem.

   You are given a list of formulas of the following form:

   - $x_i = 0$,
   - $x_i = 1$,
   - $x_i \geq x_j$ or $x_k < X$, and
   - $x_i > x_j$ or $x_k \leq X$.

   Here X denotes a set of variables, and $x_k < X$ means that $x_k < x$ for all $x \in X$; $x_k \leq X$ is defined similarly. The question is whether there exists a way to assign the values 0 and 1 to the variables such that all formulas are satisfied.

   Design a polynomial-time mapping reduction from 3-SAT to this problem.

Challenge problem

6. Given a Boolean circuit C, you want to find a Boolean circuit C' that behaves the same as C on every input but has as few gates as possible. Show that if $\mathsf{P} = \mathsf{NP}$, then this problem can be solved in polynomial time.

Programming problem

7. SPOJ problem TWOSAT (https://vn.spoj.com/problems/TWOSAT/). As this problem is only available on the Vietnamese SPOJ website, an English translation is provided on the following pages.

TWOSAT - Travel

A travel company is organizing a group of foreign tourists to travel to M cities in Vietnam. Each visitor has two requirements of the form "Want to go to c" or "Do not want to go to c", for some city c. These tourists are very difficult, and want at least one of their requirements to be met. The travel agency is struggling to choose a set of cities to visit that appeases all the tourists. You are tasked with helping them solve this problem.

Input Line 1 contains two integers, N and M ($1 \leq N \leq 20000$, $1 \leq M \leq 8000$). They are the number of tourists and cities, respectively. The cities are numbered 1 through M.

Each of the next N lines contains two integers u, v with $1 \leq |u| \leq M$ and $1 \leq |v| \leq M$, encoding the request of one of the tourists. A positive integer indicates the customer wants to go to that city, while a negative integer indicates the customer does not want to go to that city.

Output On line 1, write `YES` if there is a plan that satisfies at least one requirement of every visitor; write `NO` in the opposite case.

If there is a plan, the next two lines of output shall encode one such plan. On line 2, write a positive integer k, indicating how many cities will be visited in the plan. On line 3, write k integers, the indices of the cities to visit.

Example 1

Input:

```
2 3
-1 -2
1 2
```

Output:

```
YES
2
2 3
```

Example 2

Input:

```
3 3
-1 2
-1 -2
1 -2
```

Output:

```
YES
0
```

Example 3

Input:

```
4 3
-1 2
-1 -2
1 -2
1 2
```

Output:

```
NO
```