1. How many satisfying assignments does the following formula have?

1 / 1 point

$$(x_1 \vee \overline{x}_2 \vee \overline{x}_3)(x_1 \vee x_2)(\overline{x}_1 \vee \overline{x}_2)$$

3

Correct

That's right!

2. How many integer solutions does the following linear program have?

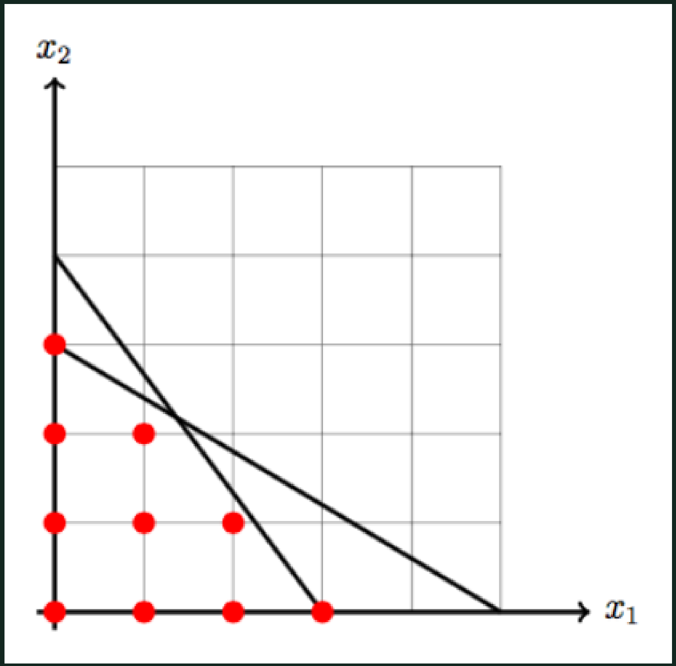
1 / 1 point

$$x_1 \geq 0, \quad x_2 \geq 0, \quad 4x_1 + 3x_2 \leq 12, \quad 3x_1 + 5x_2 \leq 15$$

10

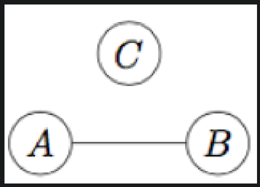
Correct

Right!



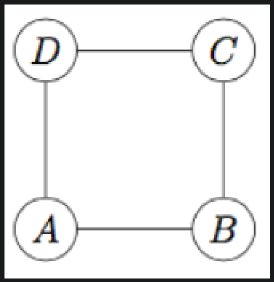
3. Consider the following graph:

1 / 1 point



It has 6 different independent sets: empty set, $\{A\}, \{B\}, \{C\}, \{A, C\}, \{B, C\}$.

How many different independent sets does the following graph have?



7

Correct

That's right! They are empty set, $\{A\}, \{B\}, \{C\}, \{D\}, \{A, C\}, \{B, D\}$.

4. In the 3-coloring problem, you are given an undirected graph and the goal is to assign one of three available colors to its vertices such that the ends of each edge of the graph receive different colors. This is clearly a search problem: given a graph and a coloring of its vertices, one can check in polynomial time whether there are only three different colors and that no edge is monochromatic. This problem is known to be NP-complete. Do we have a polynomial time algorithm for this problem?

1 / 1 point

- (x) This is an open problem.
- ( ) No, this problem cannot be solved in polynomial time for sure.
- ( ) Yes, this problem can be solved in polynomial time.

Correct

That's right! We cannot find a polynomial time algorithm for any of NP-complete problems. Nor we can prove that there is no such algorithm. Thus, the existence of a polynomial time algorithm for an NP-complete problem is an open problem.

5. In the lectures, we constructed a reduction from 3-SAT to Independent Set. Now, we show the reverse reduction. For this, we are going to reduce Independent set to SAT. We can then use the fact that SAT reduces to 3-SAT.

1 / 1 point

In the Independent Set problem we are given a graph G with n vertices $\{1, 2, \ldots, n\}$ and a positive integer b. Our goal is to check whether the graph has b vertices $\{u_1, u_2, \ldots, u_b\} \subseteq \{1, 2, \ldots, n\}$ with no edge between any pair of them. We are going to construct a CNF formula F that is satisfiable if and only if the graph G contains such an independent set. There will be bn Boolean variables: for $1 \le i \le b$ and $1 \le j \le n$, $x_{ij} = 1$ if and only if the i-th vertex of the required independent set is the j-th vertex of the graph (that is, $u_i = j$).

We then introduce the following constraints:

1. u_i is equal to some vertex of the graph: for all $1 \le i \le b$, $(x_{i1} \vee x_{i2} \vee \cdots \vee x_{in})$;
2. u_i is equal to exactly one vertex of the graph: for all $1 \le i \le b$ and all $1 \le j \ne j' \le n$, $(\overline{x}_{ij} \vee \overline{x}_{ij'})$;
3. $u_i \ne u_{i'}$: for all $1 \le i \ne i' \le b$ and all $1 \le j \le n$, $(\overline{x}_{ij} \vee \overline{x}_{i'j})$;
4. no two vertices from the independent set are joined by an edge: for all $1 \le i \ne i' \le b$ and all $\{j, j'\} \in E(G)$, $(\overline{x}_{ij} \vee \overline{x}_{i'j'})$.

The resulting formula is satisfiable if and only if the initial graph has an independent set of size b.

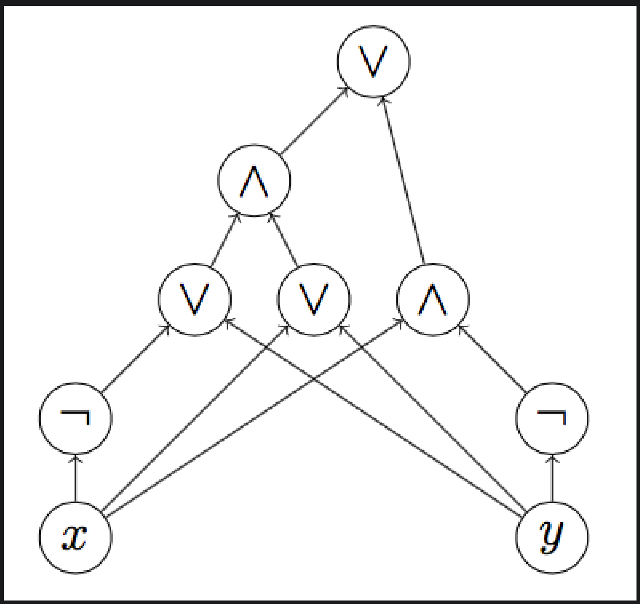
Is this reduction correct?

- ○ No, it is not correct, because it is not a polynomial time reduction.
- ◉ Yes, the reduction is correct.
- ○ No, it is not correct, because it might produce an unsatisfiable formula for a graph that has an independent set of size b.
- ○ No, it is not correct, because for a graph that does not have an independent set of size b it might produce an a satisfiable formula.

Correct
That's right!

6. How many satisfying assignments does the following circuit have?

1 / 1 point



3

Correct
That's right! The only falsifying assignment is $x = y = 0$.