Read before you start
This exam is designed so that the answers require no English words. Any English you write will not be graded. The answers are short. If you happen to write extra stuff in the designated space, make sure to clearly mark (through a circle or box) the part that you want graded.

The Exam
Problem 1: SAT Encoding
(50 points) The goal of this problem is to select a number of teachers to cover a number of subjects. This is one problem, broken into parts so that you may earn partial credits and have some guidance.

- Let \( T = \{T_1, \ldots, T_n\} \) a set of teachers.
- Let \( S = \{S_1, \ldots, S_m\} \) be a set of subjects.
- Each subject \( s \in S \) is taught by a set \( T(s) \) of the teachers \((T(s) \subseteq T)\). Additionally, you may want to make use of the function \( \text{subsets}(T, n) = \{T' \subseteq T \mid |T'| = n\} \) where \(|T|\) denotes the cardinality of a set.

Given a natural number \( k \leq n \), we want to check if it is possible to recruit at most \( k \) teachers and cover all the subjects. We use the following boolean variables:

- Teachers \( t_i \), for \( i = 1, \ldots, n \), where \( t_i \) is true if and only if the teacher is recruited.
- Subjects \( s_j \) for \( j = 1, \ldots, m \), where \( s_j \) is true if and only if there is a teacher that teaches it.

You are only allowed to use these variables in your answers to the questions below. Go through the following steps to produce the encoding for this problems:

(a) (10 points) Write a constraint (formula) that ensures all the subjects in \( S \) are taught.

\[
\bigwedge_{j=1..m} s_j
\]

(b) (15 points) Write a constraint (formula) that ensures at most \( k \) teachers are recruited. Your formula should be in CNF.

\[
\bigwedge_{c \in \text{subsets}(T, k+1)} \bigvee_{t \in c} \neg t
\]
(c) (10 points) Write a constraint (formula) that ensures that a subject is considered covered only if at least one teacher who covers it is recruited. Your formula should be in CNF.

\[ \bigwedge_{j=1}^{m} \neg s_j \lor \bigvee_{t \in T(s_j)} t \]

(d) (15 points) We have so far solved the problem of recruit at most \( k \) teachers. Now assume that we want exactly \( k \) teachers to be recruited, such that all the subjects are taught. What other constraints are required (in addition to the ones you listed in (a),(b), and (c) parts)? Write a CNF formula for the additional constraints.

\[ \bigwedge_{c \in \text{subsets}(T,n-k+1)} \bigvee_{t \in c} t \]

Problem 2: DPLL Conflict Clause Learning

(50 points) Consider a formula \( F \) that is being checked for satisfiability in CNF using the following set of clauses:

\[
\begin{align*}
    c_1 &= \neg x_2 \lor \neg x_4 \lor x_7 \\
    c_2 &= \neg x_2 \lor \neg x_7 \lor x_8 \\
    c_3 &= \neg x_5 \lor \neg x_8 \\
    c_4 &= x_4 \lor x_6 \\
    c_5 &= \neg x_1 \lor x_3 \lor x_5 \\
    c_6 &= x_2 \lor \neg x_5 \lor x_6
\end{align*}
\]

Consider a hypothetical situation where we inherit the following implication graph from decisions previously made at levels 3 and 5 of a DPLL algorithm, and we add a current decision node at the current level 6.

(a) (15 points) Complete the above implication graph (in place) for level 6 to get a conflict.
(b) (35 points) Produce an asserting clause that would be learned by the DPLL algorithm before it backtracks. To justify your answer, write down the sequence of binary resolution steps (and all the intermediate clauses) that gets you to your asserting clause. For clarity, label each clause that you use with its identifier from the list, e.g. $c_5 : \neg x_1 \lor x_3 \lor x_5$ instead of just $\neg x_1 \lor x_3 \lor x_5$.

Hint: there is more than one correct answer for an asserting clause, but the shortest one has only two literals. We will accept any asserting clause as a correct answer.

$$

c_2 : \neg x_2 \lor \neg x_7 \lor x_8 \quad c_1 : \neg x_2 \lor \neg x_4 \lor x_7 \\
\therefore \neg x_2 \lor \neg x_4 \lor x_7
$$

$$
\neg x_2 \lor \neg x_4 \lor x_7 \quad c_4 : x_4 \lor x_6 \\
\therefore \neg x_2 \lor x_6 \lor x_8
$$

$$
\neg x_2 \lor x_6 \lor x_8 \quad c_6 : x_2 \lor \neg x_5 \lor x_6 \\
\therefore \neg x_5 \lor x_6 \lor x_8
$$

It would be correct to stop here and declare $\neg x_5 \lor x_6 \lor x_8$ as your asserting clause. But, you can also continue one more step:

$$
\neg x_5 \lor x_6 \lor x_8 \quad c_3 : \neg x_5 \lor \neg x_8 \\
\therefore \neg x_5 \lor x_6
$$

to get the shorter asserting clause $\neg x_5 \lor x_6$. It is correct, but excessive to do one more step:

$$
\neg x_5 \lor x_6 \quad c_5 : \neg x_1 \lor x_3 \lor x_5 \\
\therefore \neg x_1 \lor x_3 \lor x_5
$$

and end up with $\neg x_1 \lor x_3 \lor x_5$, which is yet another asserting clause involving the three original decision literals.