

IMPORTANT: READ THE FOLLOWING DIRECTIONS.

- For each problem, write your solution using **ONE SHEET OF PAPER ONLY (BOTH FRONT AND BACK)**. Write **NAME** and **PROBLEM NUMBER** on each sheet.
- Write solutions to different problems on **SEPARATE SHEETS** of paper.

Unit 2 LO Problems

LO4. Answer the following.

(a) Consider the 3SAT instance

$$\mathcal{C} = \{c_1 = (x_1, x_2, x_2), c_2 = (\bar{x}_1, \bar{x}_2, \bar{x}_2), c_3 = (\bar{x}_1, x_2, x_2), c_4 = (x_1, \bar{x}_2, \bar{x}_2)\}.$$

and the mapping reduction $f : 3SAT \rightarrow \text{Subset Sum}$ from 3SAT to Subset Sum that was presented in lecture. Draw the complete table associated with $f(\mathcal{C}) = (S, t)$, including target t , and either provide a subset $A \subseteq S$ that sums to t or explain why one does not exist. When listing the members of A , use their corresponding variable names (i.e. y's, z's, g's, and h's).

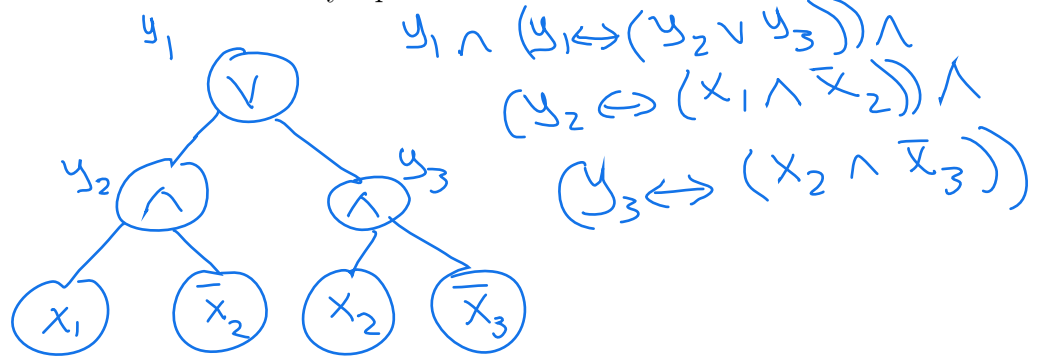
Solution. $f(\mathcal{C}) = (S, t)$ is a negative instance of Subset Sum since $\mathcal{C}$ is unsatisfiable.

	1	2	c_1	c_2	c_3	c_4
y_1	1	0	1	0	0	1
z_1	1	0	0	1	1	0
y_2	0	1	1	0	1	0
z_2	0	1	0	1	0	1
g_1			1	0	0	0
h_1			1	0	0	0
g_2				1	0	0
h_2				1	0	0
g_3					1	0
h_3					1	0
g_4						1
h_4						1
$t =$	1	1	3	3	3	3

(b) Answer the following regarding the mapping reduction from SAT to 3SAT that uses the Tseytin transformation.

- i. If the Boolean formula $F(x_1, x_2, x_3) = (x_1 \wedge \bar{x}_2) \vee (x_2 \wedge \bar{x}_3)$, then provide the initial Boolean formula that is satisfiability equivalent to F . Hint: this formula uses “double arrows”.

Solution.



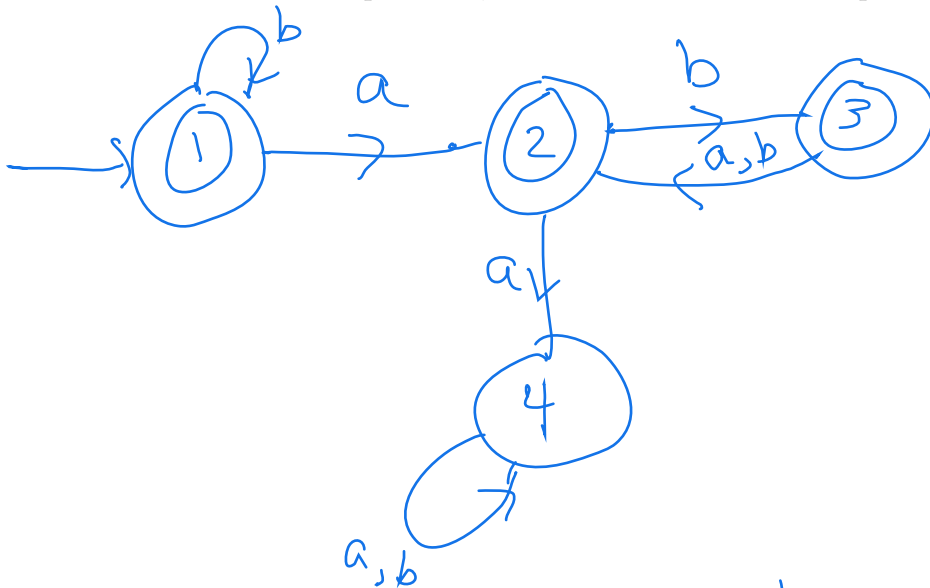
- ii. By using the steps of the Tseytin transformation described in lecture, show how to convert the Boolean formula $y \leftrightarrow (x_1 \vee \bar{x}_2)$ to a collection of 3SAT clauses.

Solution.

$$\begin{aligned}
 y \leftrightarrow (x_1 \vee \bar{x}_2) &\Leftrightarrow y \rightarrow (x_1 \vee \bar{x}_2) \wedge \\
 (x_1 \vee \bar{x}_2) \rightarrow y &\Leftrightarrow (\bar{y} \vee x_1 \vee \bar{x}_2) \wedge \\
 \overline{(x_1 \vee \bar{x}_2)} \vee y &\Leftrightarrow [\bar{x}_1 \wedge x_2 \vee y] \wedge (\bar{y} \vee x_1 \vee \bar{x}_2) \\
 &\Leftrightarrow (\bar{x}_1 \vee y) \wedge (x_2 \vee y) \wedge (\bar{y} \vee x_1 \vee \bar{x}_2) \\
 &\Rightarrow \text{clauses are } (\bar{x}_1, y, y), (x_2, y, y), \\
 &\quad (\bar{y}, x_1, \bar{x}_2).
 \end{aligned}$$

LO5. Do the following.

- (a) Consider the language L of all words w over the alphabet $\{a,b\}$ for which one of the following is true: i) w has at most one a or ii) w has at least two a 's and, for any two a 's for which there are no other a 's between them, there must be an odd number of b 's between them. For example $abbba$ and $ababbbba$ are two words that satisfy ii). Provide a DFA M that accepts L .
- (b) Demonstrate the computation of M on inputs i) $w_1 = bbabbbba$ and ii) $w_2 = ababbab$. For each computation, indicate whether w is accepted or rejected.



b b a b b b a 2
 1 1 1 2 3 2 3 2

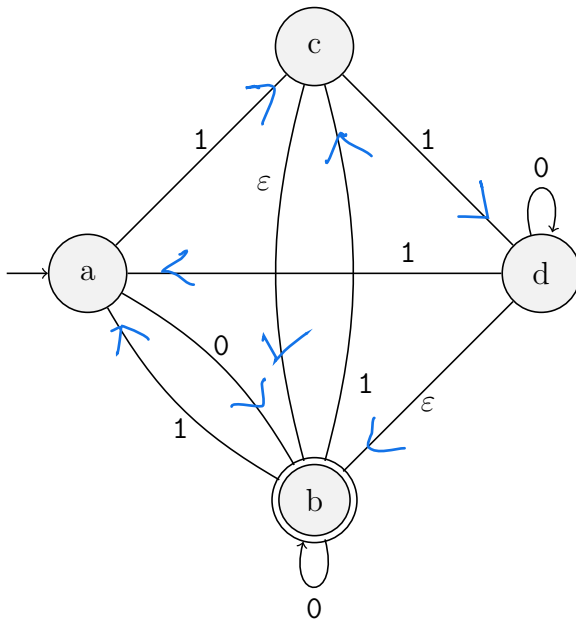
2
 $\Downarrow$
 accept

a b a b b a b
 1 2 3 2 3 2 4

4
 $\Downarrow$
 reject

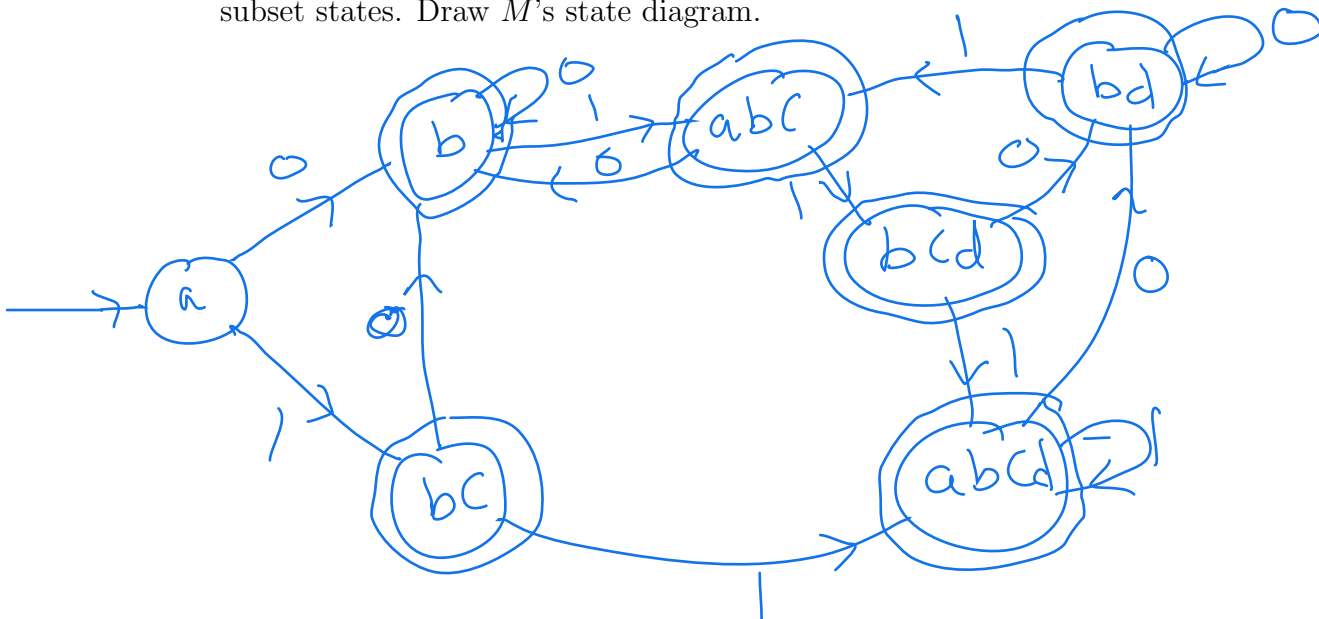
LO6. Do the following.

- (a) For the NFA N whose state diagram is shown below, provide a state transition table for N .



	0	1
a	b	bc
b	b	abc
c	$\emptyset$	bd
d	bd	a

- (b) Use the table from part b to convert N to an equivalent DFA M using the method of subset states. Draw M 's state diagram.



Makeup Problems

LO1. Consider the 2SAT instance

$$\mathcal{C} = \{(x_1, x_3), (\bar{x}_1, x_5), (x_2, x_4), (\bar{x}_2, \bar{x}_3), (\bar{x}_2, x_4), (\bar{x}_2, \bar{x}_4), (\bar{x}_3, x_4), (\bar{x}_3, x_5), (\bar{x}_4, x_6), (\bar{x}_4, \bar{x}_6)\}.$$

- (a) Draw the implication graph $G_{\mathcal{C}}$.

Solution.

- (b) Perform the **Improved 2SAT** algorithm by computing the necessary reachability sets. Use numerical order (in terms of the variable index) and positive literal before negative literal when choosing the reachability set to compute next. Draw the resulting reduced 2SAT instance whenever a consistent reachability set is computed. Either provide a final satisfying assignment for $\mathcal{C}$ or indicate why $\mathcal{C}$ is unsatisfiable.

Solution.

- (c) Suppose 2SAT instance $\mathcal{C}$ is satisfiable and has the unique satisfying assignment $\alpha = (x_1 = 1, x_2 = 0, x_3 = 1)$. If the original 2SAT algorithm is run with $\mathcal{C}$ as input, then at most how many queries will the algorithm make to the **Reachability** oracle? Explain. Hint. For each i , assume that the query $\text{reachable}(G_{\mathcal{C}}, x_i, \bar{x}_i)$ precedes the query $\text{reachable}(G_{\mathcal{C}}, \bar{x}_i, x_i)$.

LO2. Do the following.

- (a) Provide the definition of what it means to be a mapping reduction from problem *A* to problem *B*. Hint: do *not* assume that *A* and *B* are decision problems.
- (b) Is $f(n) = 4n^2 - 3n + 7$ a valid mapping reduction from **Even** to **Odd**? If yes, then defend your answer by providing a general argument based on how arithmetic operations affect even and odd numbers. Simply plugging in a few numbers is unacceptable. However, if f is *invalid*, then give a specific example that supports this claim.

Solution. We use the following fact about even and odd integers.

$$\text{even} \times I = \text{even}$$

where I is any integer,

$$\text{odd} \times \text{odd} = \text{odd}$$

$$\text{even} + \text{even} = \text{odd} + \text{odd} = \text{even},$$

and

$$\text{even} + \text{odd} = \text{odd}.$$

Then if n is even, then

$$f(\text{even}) = \text{even} \times (\text{even})^2 + \text{odd} \times \text{even} + \text{odd} = \text{even} + \text{even} + \text{odd} = \text{odd},$$

and so a positive instance of **Even** maps to a positive instance of **Odd**. Similarly, if n is odd, then

$$f(\text{odd}) = \text{even} \times (\text{odd})^2 + \text{odd} \times \text{odd} + \text{odd} = \text{even} + \text{odd} + \text{odd} = \text{even},$$

and so a negative instance of **Even** maps to a negative instance of **Odd**. Therefore, f is a valid mapping reduction from **Even** to **Odd**. $\square$

LO3. An instance of **Dominating Set** (DS) is a simple graph $G = (V, E)$ and a natural number $k \geq 0$. The problem is to decide if there is a set D of k vertices of G for which every vertex *not* in D is adjacent to some vertex in D . To see that DS is an NP problem we define a certificate for instance (G, k) to be a set $D \subseteq V$ of k vertices. The following pseudocode is used by the verifier to determine if D is in fact a dominating set for G . Note: a minimum of 18 points is needed to pass this LO.

For each $u \in V$,

 If $u \in D$, then continue. //Skip any vertex that is in D .

 found = 0. //We have yet to find a vertex $v \in D$ for which $(u, v) \in E$.

 For each $v \in D$,

 If $(u, v) \in E$,

 · found = 1 //found a vertex $v \in D$ for which $(u, v) \in E$.

· Break out of the inner loop.

If found = 0, return 0.

Return 1.

- (a) Provide size parameters for the DS problem and describe what each represents in relation to a DS problem instance. Hint: there are two of them. (6 pts)

Solution. $m = |E|$ and $n = |V|$.

- (b) Use the size parameters to provide the big-O number of steps that is required by the verifier to check the validity of a certificate. Justify your answer. (7 pts)

Solution. The outer loop iterates at most n times and, for each such iteration, the inner loop also iterates at most n times. Checking if an ordered pair is an edge of the graph requires $O(1)$ steps assuming the use of a hash table which requires $O(m)$ steps to create. Therefore, the number of steps equals $O(n^2 + m) = O(n^2)$ and the verifier requires a quadratic number of steps.

- (c) Classify each of the following problems as being in P, NP, or co-NP (3 points each).

Solution. P, co-NP, co-NP, NP

- i. An instance of the **3-Dimensional Matching (3DM)** decision problem takes as input three sets A , B , and C , each having size n , along with a set S of m triples of the form (a, b, c) where $a \in A$, $b \in B$, and $c \in C$. The problem is to decide if there is a subset $T \subseteq S$ of n triples for which each member from $A \cup B \cup C$ belongs to exactly one of the triples.
- ii. An instance of **Fifty Subset Sum (FSS)** is a pair (S, t) , where S is a set of natural numbers and $t \geq 0$ is a natural number. The problem is to decide if there are exactly 50 members of S whose sum adds to t .
- iii. An instance of **Tree** is a simple graph $G = (V, E)$, and the problem is to decide if G is a tree, meaning that it has no cycles (of length three or more) and it is connected, meaning that there is path from any one vertex to another.
- iv. An instance of **Tautology** is a Boolean formula $F(x_1, \dots, x_n)$ and the problem is to decide if F is a tautology, meaning that $F(\alpha) = 1$ for every assignment α over the variables $x_1, \dots, x_n$.