CECS 329, Exam 2 Friday Spring 2025, Dr. Ebert

IMPORTANT: READ THE FOLLOWING DIRECTIONS. Directions,

- For each of LO's 4-8 and the Additional problem, write your solution using a **SINGLE SHEET OF PAPER ONLY (BOTH FRONT AND BACK)**. Write **NAME** and **PROBLEM NUMBER** on each sheet.
- For Unit-1 LO's it's OK to use the same sheet for two or more problems if there is sufficient space.

Unit-2 LO's (25 Points Each)

LO4. Do the following.

a. Consider the `3SAT` instance

$$\mathcal{C} = \{c_1 = (x_1, x_2, x_3), c_2 = (\overline{x}_1, \overline{x}_2, \overline{x}_3), c_3 = (\overline{x}_1, x_2, x_3), c_4 = (x_1, \overline{x}_2, \overline{x}_3)\}.$$

Consider the mapping reduction f : `3SAT` $\rightarrow$ `Clique` from `3SAT` to `Clique` that was presented in lecture.

i. How many vertices and edges does G have if $f(\mathcal{C}) = (G, k)$? Show all work. Also, what is the value of k? (10 pts)

ii. Verify that $\mathcal{C}$ is satisfiable and use the satisfying assignment α to provide a corresponding k-clique for G. For each vertex of the clique, indicate it's vertex group. (7 pts)

b. In accordance with the Tseytin transformation from `SAT` to `3SAT`, show the steps needed to transform the Boolean formula $y \leftrightarrow (x_1 \wedge \overline{x}_2)$ to a collection of `3SAT` clauses. (8 pts)

LO5. Do the following.

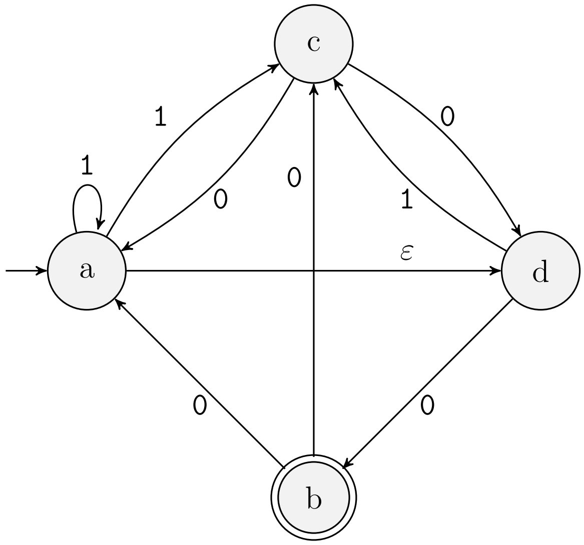
a. Let L be the language of binary words that contain at least two occurrences of the pattern 101. For example, 0010101 and 010111011 are both in the language, but 010 and 111010 are not in the language. Provide the state diagram for a DFA M that accepts L. (12 pts)

b. For each state s that appears in your DFA from part a, provide a sentence that describes what must be true about *any* word that, when read by M causes M to finish its computation in state s. (7 pts)

c. Demonstrate the computation of M on inputs i) $w_1 = 10010101$ and ii) $w_2 = 101110$. For each computation, indicate whether w is accepted or rejected. (6 pts)

LO6. For the NFA N whose state diagram is shown below,

a. provide a table that represents N's δ transition function. (12 pts)



b. Use the table from part a to convert N to an equivalent DFA M using the method of subset states. Draw M's state diagram. (8 pts)

c. Show the computation of M on input 1011001. (5 pts)

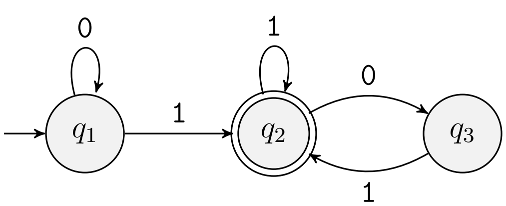
LO7. Do the following.

a. Let L_1 denote the language of binary words that contain exactly two 0's and an odd number of 1's and, L_2 denote the language consisting of all binary words w that have at least two 0's and at most two 1's. Using De Morgan's rule to compute $\overline{L}_1$ we get $\overline{L}_1 = A \cup B$. Provide descriptions of the A and B languages. (6 pts)

b. Is 101011100011 a member of L_1L_2? Explain. (6 pts)

c. Provide a regular expression that represents the language consisting of all binary words that do *not* contain substring 010. (13 pts)

LO8. Consider the NFA N shown below.



and let L denote the language that it recognizes.

a. Use N to construct the NFA N' that accepts L^* and uses the algorithm that converts an NFA that accepts L, to one that accepts L^*. (10 pts)

b. Demonstrate each step of the GNFA-to-Regular-Expression algorithm that computes a regular expression that describes L. Hint: your initial GNFA should have five states. (15 pts)

Additional Problem

Answer the following.

1. What does it mean for a decision problem to be NP-Complete? Provide the official definition. (10 pts)
2. Provide the chain of mapping reductions that establishes the NP-completeness of the `Vertex Cover (VC)` decision problem. Hint: `Independent Set` $\leq_m^p$ `VC` is the final link of the chain. (8 pts)
3. Professor Jones has just discovered a polynomial-step algorithm for deciding any instance of the `Subset Summ` decision problem. Explain why this proves that P = NP (12 pts)

Unit-1 LO's (0 Points Each)

LO1. Consider the 2SAT instance

$$\mathcal{C} = \{(\overline{x}_1, x_2), (x_1, x_5), (\overline{x}_2, x_3), (x_2, x_3), (x_2, \overline{x}_5)(\overline{x}_3, x_4), (\overline{x}_3, \overline{x}_4), (x_4, \overline{x}_5)\}.$$

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

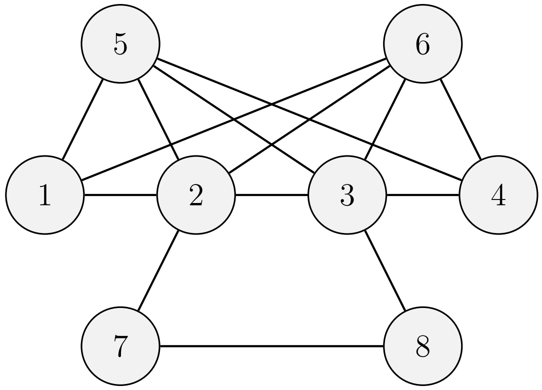
(b) Perform the `Improved 2SAT` algorithm by computing the necessary reachability sets. Use numerical order (in terms of the variable index) and positive literal before negative literal when choosing the reachability set to compute next. Draw the resulting reduced 2SAT instance whenever a consistent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

(c) If an instance $\mathcal{C}$ has 336 variables and 2027 clauses, then what is the least number of queries that must be made to the `Reachability` oracle in order to confirm that $\mathcal{C}$ is satisfiable? Explain.

LO2. Answer the following.

(a) Provide the definition of what it means to be a mapping reduction from decision problem A to decision problem B.

(b) For the mapping reduction f : `Clique` $\rightarrow$ `Half Clique`, draw $f(G, k)$ for the `Clique` instance whose graph is shown below, and for which $k = 5$.



(c) Verify that both (G, k) and $f(G, k)$ are either both positive instances, or are both negative ones. Explain and show work.

LO3. An instance of `Mine Sweep` is a simple graph $G = (V, E, f)$ where $f : V \to \{-1, 0, 1, \ldots\}$ is a function from the set of vertices to the set of natural numbers, including -1. If $f(v) \geq 0$, then it means that a total of $f(v)$ neighbors of v (i.e., vertices that are adjacent to v) must have a mine placed on them. On the other hand, if $f(v) = -1$, then there is no constraint on how many neighbors of v must have a mine. The problem is to decide if there is a function $g : V \to \{0, 1\}$, such that i) $g(v)$ indicates whether or not a mine is placed on vertex v, and ii) for all $v \in V$, if $f(v) \geq 0$, then

$$f(v) = \sum_{u \in N(v)} g(u),$$

where $N(v)$ is the set of all neighobrs of v. In other words, function g meets all the mine constraints that are indicated by f. To see that `Mine Sweep` is an `NP` problem we define a certificate for instance G to be a function $g : V \to \{0, 1\}$. The following pseudocode is used by the verifier to determine if g is in fact a valid function that indicates where mines should and should not be placed. Note: a minimum of 18 points is needed to pass this LO.

For each $u \in V$,

If $f(u) == -1$, then continue. //vertex u does not create a constraint on its neighbors

sum $= 0$. //initialize sum

For each $v \in N(u)$,

If $g(v) == 1$, then sum $++$.

If $f(u) \neq$ sum, then return 0. g is an invalid mine-placement function

Return 1.

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

(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. number of steps with respect to the size parameters of G. Justify your answer. Hint: we may assume that addition is O(1) and both f and g are represented as arrays. (7 pts)

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

i. An instance of `Unbalanced Set` is a Set S of natural numbers. The problem is to decide if for any subset $A \subseteq S$, the sum of A's members does not equal the sum of the members of its complement.

ii. The `Vertex Cover` decision problem studied in the `Mapping Reducibility` lecture.

iii. An instance of `Tautology` is a Boolean formula $F(x_1, \ldots, x_n)$ and the problem is to decide if $F(\alpha) = 1$ for every assignment α over the variables $x_1, \ldots, x_n$.

iv. An instance of `Palindrome` is an array a of integer, and the problem is to decide if a reads the same forwards as backwards, meaning that, for all $i \in \{0, 1, \ldots, n-1\}$, $a[i] = a[n-1-i]$.

v. An instance of the `One Hundred Club` is a graph $G = (V, E)$, and the problem is to decide if there is a subset C of one-hundred vertices of G for which every vertex not in C is adjacent to at least one of the vertices in C.