CECS 329, Quiz 4, Fall 2025, Dr. Ebert

IMPORTANT: READ THE FOLLOWING DIRECTIONS. Directions,

- For LO's 1-10 please solve each on a **SINGLE SEPARATE SHEET OF PAPER ONLY (BOTH FRONT AND BACK)**. Write **NAME** and **PROBLEM NUMBER** on each sheet.
- **TEN STACKS OF PAPERS: LO's 1-10**

Learning Outcome Assessment Problems

LO10. Do the following.

(a) The following is the δ-transition table for a Turing machine M. Show the first five configurations of M's computation on input word 101. Note: a is the initial state.

Q/Γ	0	1	$\sqcup$	x
a	b,x,R	c,x,R	n/a	n/a
b	b,0,R	b,1,R	d,$\sqcup$,L	d,x,L
c	c,0,R	c,1,R	e,$\sqcup$,L	e,x,L
d	f,x,L	n/a	n/a	f,x,L
e	n/a	f,x,L	n/a	f,x,L
f	f,0,L	f,1,L	n/a	a,x,R

(b) Provide the state diagram for a Turing machine M that has the following behavior. On (nonempty) ternary input word w, M checks that there is no 1 that is somewhere to the left of any 0. If there is, then M immediately rejects w. Otherwise, M moves to the location of the first 1 in w and accepts. For example, M should accept 002022121 and the head's final resting place will be at the cell holding the first 1. However, M should reject 202012202. Hint: you may find it helpful to include x in your tape alphabet.

LO9. Do the following.

(a) Use the CFG rules below to derive the expression $(a+a)\times a$. Make sure to use a left-most deriviation and replace at most one variable in each step.

$$E \to E + T \mid T$$

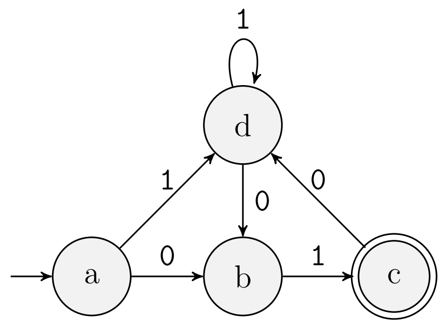
$$T \to T \times F \mid F$$

$$F \to (E) \mid a$$

(b) Provide the rules for a context-free grammar whose terminal set is $\Sigma = \{a, b, c\}$ and which derives exactly those words having the form $a^m b^n c^p$, where $m < p$ and there is no constraint on the number of b's.

LO8. Do the following.

(a) Consider the NFA N shown below.



and let L_1 denote the language that it recognizes. Let L_2 denote the language consisting of all words that have an odd number of 1's. Use the technique presented in lecture to provide a state diagram of an NFA that accepts the language $L_1 \circ L_2$.

(b) Demonstrate each step of the GNFA-to-Regular-Expression algorithm that computes a regular expression that describes the language accepted by the NFA from part a.

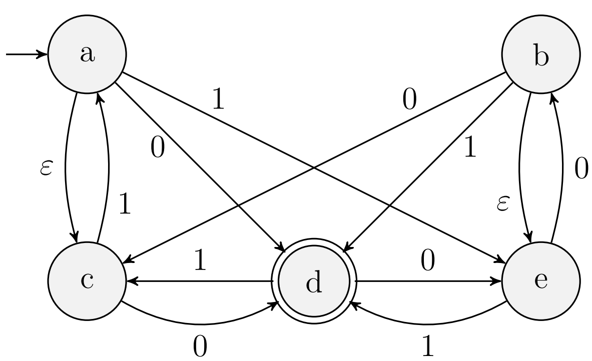
LO7. Do the following.

(a) Let L_1 denote the language of binary words that contain exactly one 1, and L_2 denote the language consisting of all binary words w that have at least three 0's. Use set notation to write the members of $L_1 - L_2$.

(b) Explain why 010001100110 a member of $(L_1 - L_2)^*$.

(c) Provide a regular expression that describes the language consisting of all binary words w for which either i) w has at most one 1 or ii) w has two or more 1's and there is always an even number of 0's between any two consecutive 1's. For example, 01 and 00010010000100 are in the language, but 00100010 is not since there are three 0's between two consecutive 1's.

LO6. For the NFA N whose state diagram is shown below, provide a table that represents N's δ transition function.



(a) Provide a table that represents N's δ transition function.

(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.

(c) Show the computation of M on input $w = 11001$.

LO5. Do the following.

(a) Let L be the language of binary words that either have at least one 0 or (*exclusive*) exactly one 1. In other words, if A is the language of all binary words that have at least one 0 and B is the binary language of all words having exactly one 1, then $L = A \oplus B$. For example, 1 is in the language since it has property B but not A. However, 11 is not in the language since it has neither property A nor property B. Also, 0010011 is in the language since it has property A but not B, but 00100 is not in the language since it has both properties. Provide the state diagram for a DFA M that accepts L.

(b) Demonstrate the computation of M on inputs i) $w_1 = 01101$ and ii) $w_2 = 0001$. For each computation, indicate whether w is accepted or rejected.

LO4. Answer the following.

(a) Consider the instance of `3SAT` instance

$$\mathcal{C} = \{c_1 = (\overline{x}_1, \overline{x}_3, x_4), c_2 = (x_2, \overline{x}_3, x_4), c_3 = (\overline{x}_2, \overline{x}_3, \overline{x}_4), c_4 = (x_1, x_2, \overline{x}_3))\}$$

and the mapping reduction f from `3SAT` to `Clique`. Answer the following with respect to $f(\mathcal{C}) = (G, k)$.

i. How many vertices and edges does G have? Explain and Show work.

ii. What is the value of k? Explain.

iii. Verify that G has a k-clique. For each vertex in the clique, indicate the vertex group to which it belongs. What does this clique tell you about $\mathcal{C}$?

(b) Given the Boolean formula $F = x_1 \wedge (\overline{x}_2 \vee x_3)$, do the following. Provide the corresponding Boolean formula that is satisfiability-equivalent to F and is the starting point of the Tseytin transformation. Hint: the new formula has both x and y variables. Show how Tseytin converts the *first* double-arrow subformula (of the formula you wrote) into a set of `3SAT` clauses.

LO3. Answer the following.

(a) An instance of the `Max Cut` decision problem is a simple graph $G = (V, E)$ and a nonnegative integer $k \geq 0$. The problem is to decide if there is a function $f : V \to \{0, 1\}$ for which

$$\sum_{e=(u,v)\in E} (f(u) \oplus f(v)) \geq k.$$

In other words, there are at least k edges for which the two vertices of the edge have been assigned opposite values by f. To establish that `Max Cut` is an `NP` problem, we define a certificate to be a function $f : V \to \{0, 1\}$. Below is the pseudocode for a verifier program that establishes that `Max Cut` is a member of `NP`. Note: a minimum of 18 points is needed to pass this LO.

sum $= 0$

For each edge $e = (u, v) \in E$,

  If $f(u) \neq f(v)$, then sum $=$ sum $+ 1$.

Return (sum $\geq k$).

i. Describe the size parameters for `Max Cut`, including what each represents. Hint: there are two of them. (6 points)

ii. Provide the big-O number of steps that is required to execute the above psuedocode. Hint: assume that it takes O(1) steps to evaluate f at any vertex. Defend your answer. (7 pts)

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

i. An instance of `Balanced Subset` is a set S of integers, and the problem is to decide if, for every member of S, it's additive inverse is also a member of S.

ii. An instance of `Boolean Subset Sum` is a set S of Boolean vectors of some common length, and a natural number $k \geq 0$, and the problem is to decide if there is a subset $A \subseteq S$ of size k for which the bitwise OR of the members of A is equal to 1.

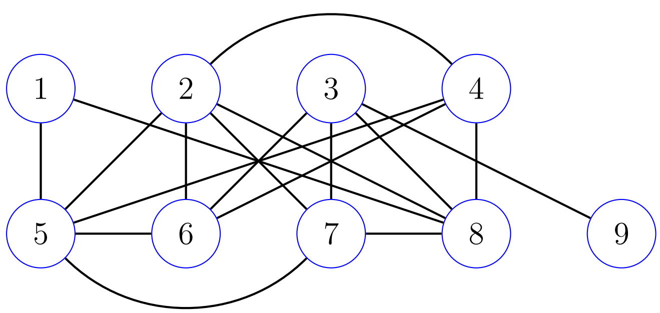
iii. An instance of `Long Increasing Subsequence` is a length-n array a of integers and a natural number $k \geq 0$, and the problem is to decide if there are indices $0 \leq i_1 \leq \cdots \leq i_k < n$ for which $a[i_1] \leq a[i_2] \leq \cdots \leq a[i_k]$.

iv. An instance of `Sum Avoidance` is a finite subset of integers S, and a target value t, and the problem is to decide if no subset of S has the property that its members sum to t.

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) The simple graph $G = (V, E)$ shown below is an instance of the `Clique` problem with $k = 4$. Draw $f(G, k)$, where f is the mapping reduction from `Clique` to `Half Clique` provided in the Mapping Reducibility lecture exercises.



(c) Verify that both G and $f(G, k)$ are either both positive or both negative instances of their respective problems. Justify your answer.

LO-1. Consider the `2SAT` instance

$$\mathcal{C} = \{(x_1, \overline{x}_2), (x_1, x_5), (x_1, x_2), (\overline{x}_1, x_4), (\overline{x}_1, x_6), (\overline{x}_2, \overline{x}_5), (x_3, x_4), (\overline{x}_3, \overline{x}_6), (\overline{x}_4, \overline{x}_5)\}.$$

(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) Suppose `2SAT` instance $\mathcal{C}$ has three variables and, when running the original `2SAT` algorithm the answer to each oracle query is shown in the table below. Is $\mathcal{C}$ satisfiable? If yes, provide a satisfying assignment for $\mathcal{C}$. If not, explain why.

Oracle Query	**Answer**
`reachable`$(G_{\mathcal{C}}, x_1, \overline{x}_1)$	Yes
`reachable`$(G_{\mathcal{C}}, \overline{x}_1, x_1)$	No
`reachable`$(G_{\mathcal{C}}, x_2, \overline{x}_2)$	Yes
`reachable`$(G_{\mathcal{C}}, \overline{x}_2, x_2)$	Yes
`reachable`$(G_{\mathcal{C}}, x_3, \overline{x}_3)$	No
`reachable`$(G_{\mathcal{C}}, \overline{x_3}, x_3)$	Yes