CECS 329, Quiz 4, Fall 2025 Friday Section, 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.

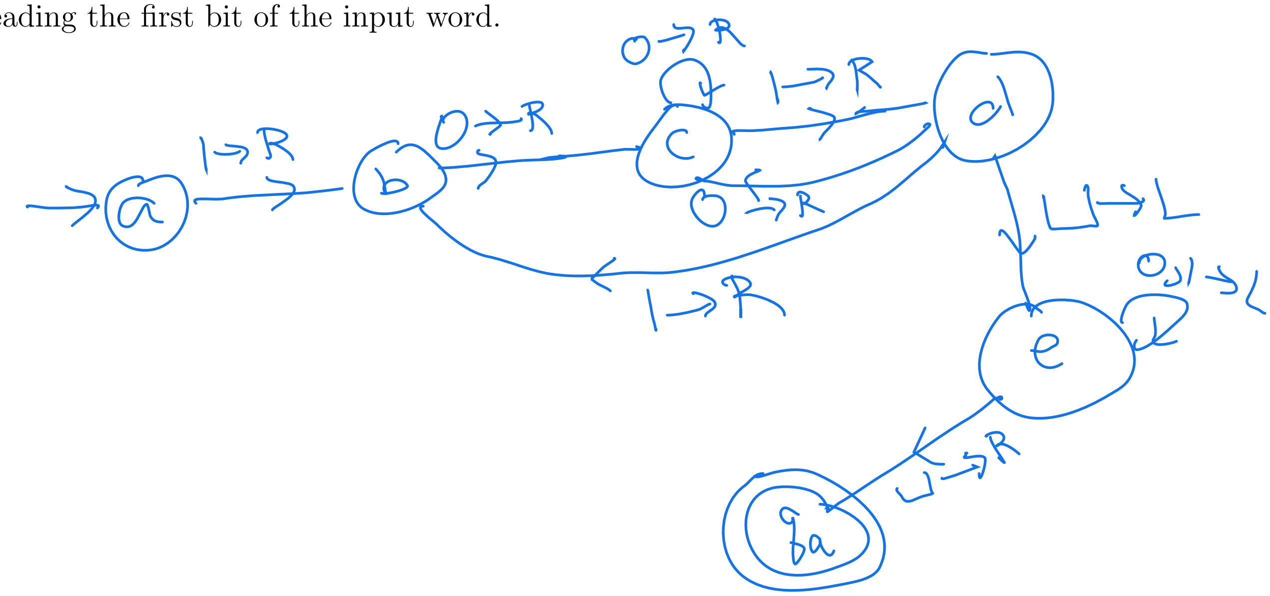
(a) The following is the δ-transition table for a Turing machine M. Show the first five configurations of M's computation on input word 101. Note: a is the initial state.

Q/Γ	0	1	⊔	x
a	b,x,R	c,x,R	n/a	n/a
b	b,0,R	b,1,R	d,⊔,L	d,x,L
c	c,0,R	c,1,R	e,⊔,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

Solution. We have

$$a101 \Rightarrow xc01 \Rightarrow x0c1 \Rightarrow x01c\sqcup \Rightarrow x0e1.$$

(b) Provide the state diagram for a Turing machine M that accepts all binary words that begin with 10 and end with 01. Moreover, when the machine halts, the head should be reading the first bit of the input word.



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.

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

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

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

Solution. We have

$$E \to T \to T \times F \to F \times F \times (E) \times F \to (E+T) \times F$$

$$(T+T) \times F \to (F+T) \times F \to (a+T) \times F \to (a+F) \times F \to$$

$$(a+a) \times F \to (a+a) \times a.$$

(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 the number of b's must be even (this includes having 0 b's).

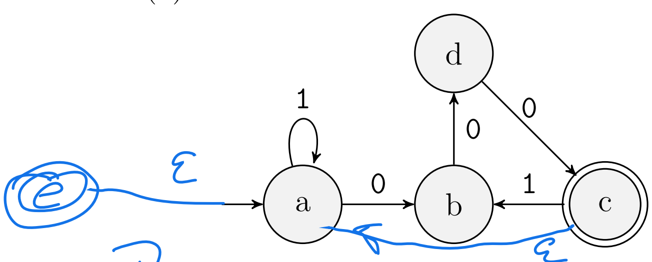
$$S \to aT.$$

$$T \to aTc \mid aT \mid B \mid \varepsilon$$

$$B \to bbB \mid \varepsilon.$$

LO8. Do the following.

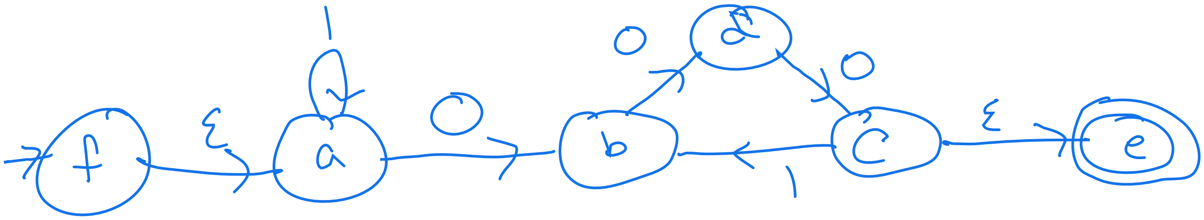
(a) Consider the NFA N shown below.



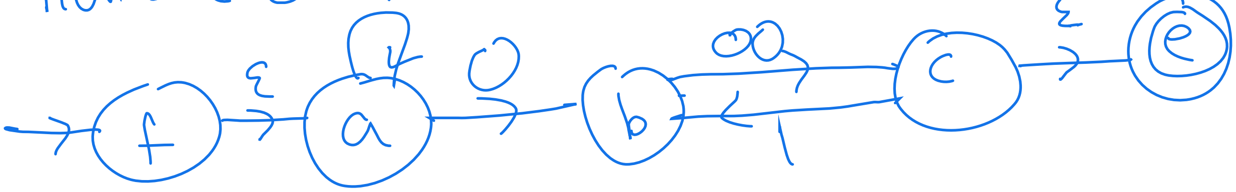
and let L_1 denote the language that it recognizes. Use the technique presented in lecture to provide a state diagram of an NFA that accepts the language L_1^*.

Solution.

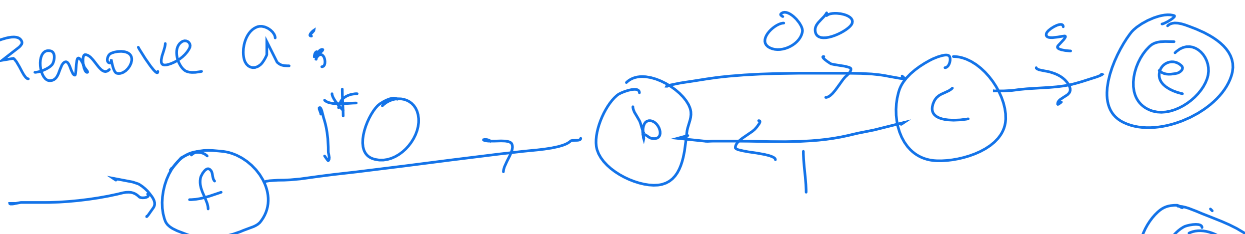
(b) Demonstrate each step of the GNFA-to-Regular-Expression algorithm that computes a regular expression that describes the language accepted by the NFA from part a.



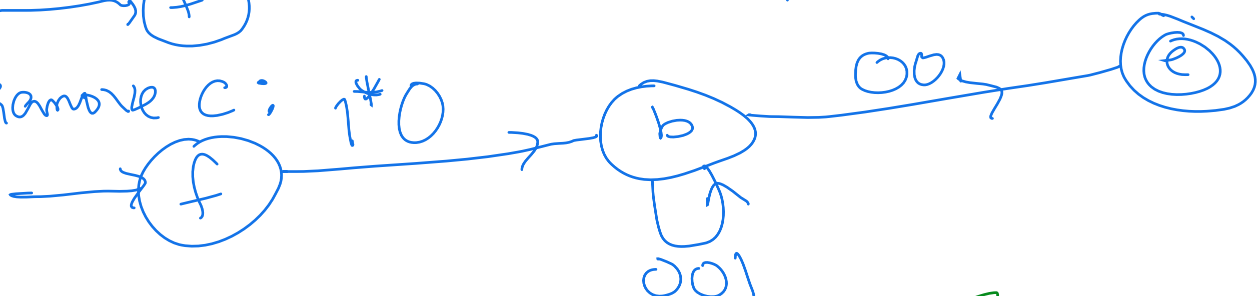
Remove d:



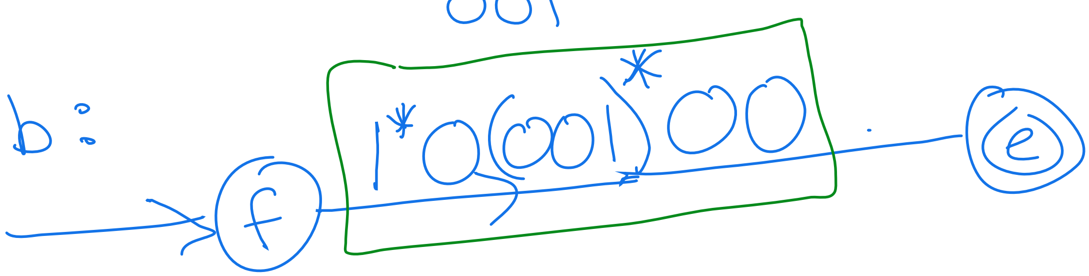
Remove a:



Ramove c:



Remove b:



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 $L_1 - L_2$.

Solution. $L_1 - L_2 = \{001, 010, 001, 01, 10, 1\}$.

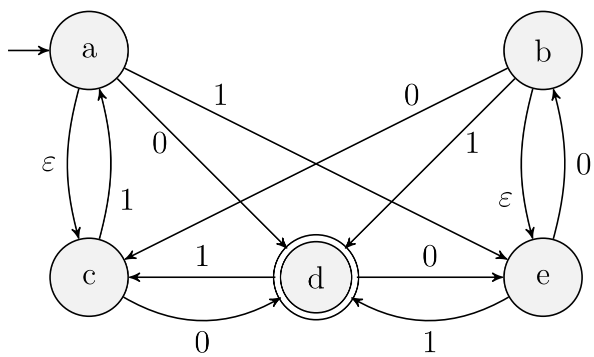
(b) Explain why 010001100110 a member of $(L_1 - L_2)^*$.

Solution. $010001100110 = 010 \cdot 001 \cdot 100 \cdot 1 \cdot 10$ if the concatentation of five words that all belong to $(L_1 - L_2)^*$.

(c) Provide a regular expression that describes the language consisting of all binary words for which every odd bit is 1 and ends with a 0.

Solution. $\{10, 11\}10$

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



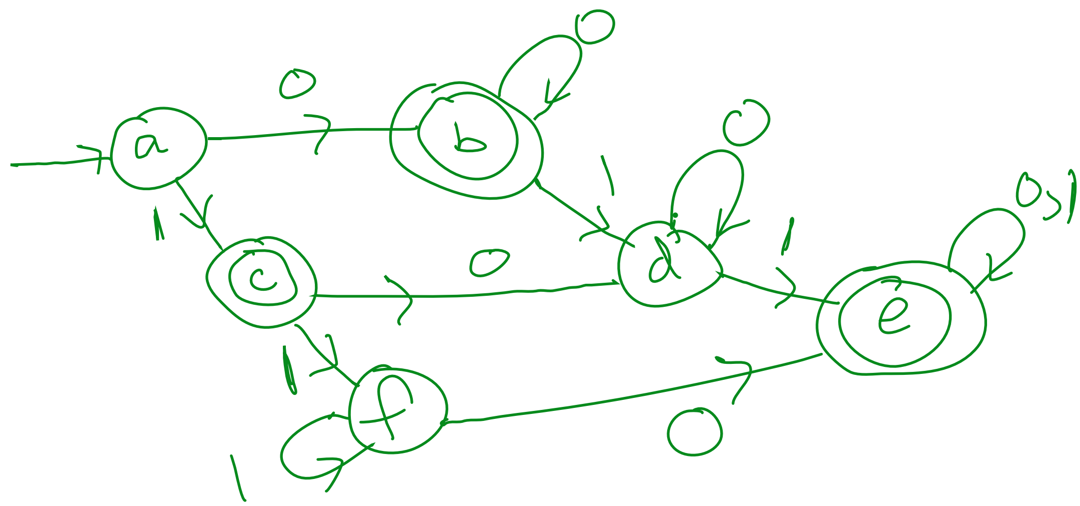
(a) Provide a table that represents N's δ transition function.

(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.

0 1 1 0 1

a b d e e $\boxed{e}$ ⇒ accept

0 0 0 1

a b b b $\boxed{d}$ ⇒ reject

LO4. Answer the following.

(a) An instance $\mathcal{C}$ of 3SAT consists of clauses $c_1 = (x_1, \overline{x}_2, x_3)$, $c_2 = (\overline{x}_2, x_3, x_4)$, $c_3 = (\overline{x}_1, x_2, \overline{x}_4)$, and $c_4 = (\overline{x}_1, \overline{x}_3, x_4)$. Answer the following questions about the mapping reduction $f(\mathcal{C}) = (G, a, b)$ provided in lecture from 3SAT to Directed Hamilton Path and applied to instance $\mathcal{C}$. Note: correctly solving the following parts counts for passing LO11.

i. Consider the vertices lc_2 and rc_2 that lie in the x_2-diamond of G as well as the c_2 vertex. Provide all the edges that exist between these three vertices. Draw a figure with the three vertices and the two edges and explain why you chose the particular orientations.

ii. Which diamond(s) has no edges connected to vertex c_2? Explain.

iii. Given that $\alpha = (x_1 = x_2 = 0, x_3 = 1, x_4 = 0)$ satisfies $\mathcal{C}$, provide the direction (left-to-right or right-to-left) that the corresponding Hamilton Path will take through each of the four diamonds of G and, for each clause vertex c, provide the diamond from which c will be visited.

(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) ∈ E,
    If f(u) ≠ f(v), then sum = sum + 1.
Return (sum ≥ 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.

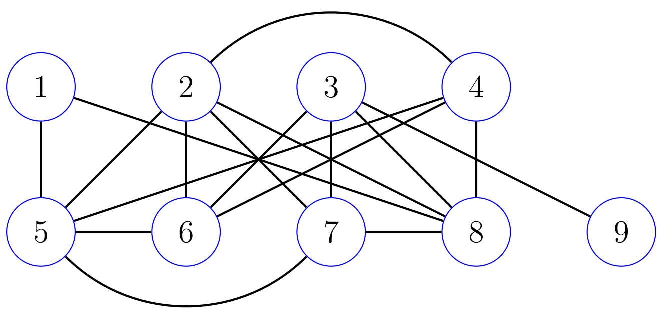
iii. An instance of `Long Increasing Subsequence` is a length-n array a of integers and a natural number $k \geq 0$, and the problem is to decide if there are indices $0 \leq i_1 \leq \cdots \leq i_k < n$ for which $a[i_1] \leq a[i_2] \leq \cdots \leq a[i_k]$.

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. (10 pts)

(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) If an instance $\mathcal{C}$ has 336 variables and 2027 clauses, then, when running the original `2SAT` algorithm, what is the least number of total queries that must be made to the `Reachability` oracle in order to confirm that $\mathcal{C}$ is unsatisfiable? Explain.