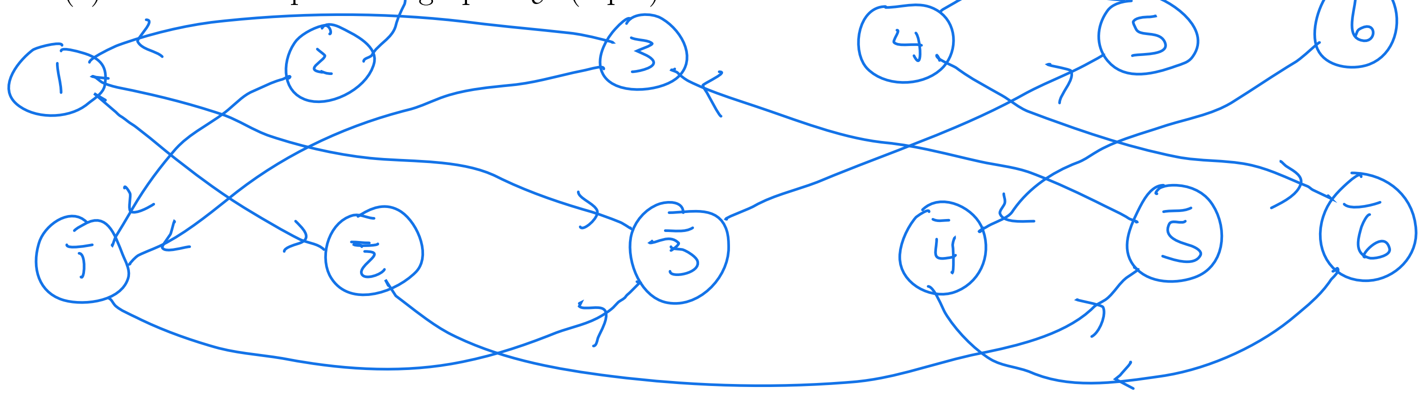
CECS 528, Exam 2, Spring 2025, Dr. Ebert

Unit 2 LO Problems (25 pts each)

LO5. Consider the 2SAT instance

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

(a) Draw the implication graph $G_{\mathcal{C}}$. (7 pts)

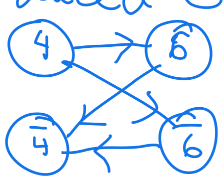


(b) Perform the `Improved 2SAT` algorithm by computing the necessary reachability sets. Use numerical order (in terms of the variable index) and positive literal before negative literal when choosing the reachability set to compute next. Draw the resulting reduced implication graph whenever a consistent reachability set is computed. Either provide a final satisfying assignment for $\mathcal{C}$ or indicate why $\mathcal{C}$ is unsatisfiable. (10 pts)

$R_{x_1} = \{x_1, \overline{x}_2, \overline{x}_3, x_5, \overline{x}_5, \ldots, \overline{x}_1\}$ is inconsistent

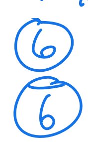
$R_{\overline{x}_1} = \{\overline{x}_1, \overline{x}_3, x_5, x_2\}$ is consistent. $\alpha_{R_{\overline{x}_1}} = (x_1=0, x_2=1, x_3=0, x_5=1)$

Reduced Graph:



$R_{x_4} = \{x_4, \overline{x}_4, \overline{x}_6\}$ is inconsistent $R_{\overline{x}_4} = \{\overline{x}_4\}$

$\alpha_{R_{\overline{x}_4}} = (x_4 = 0)$ Reduced Graph:



$\alpha_{R_{x_6}} = \{x_6 = 1\}$

(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 $\mathcal{C}$ and justify your answer. If not, explain why. (8 pts)

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

$\therefore$ $\alpha = \alpha_{R_{\overline{x}_1}} \cup \alpha_{R_{\overline{x}_4}} \cup \alpha_{R_{x_6}} = (x_1=0, x_2=1, x_3=0, x_4=0, x_5=1, x_6=1)$ satisfies $\mathcal{C}$.

Yes, since the query answers imply no inconsistent cycles in $G_{\mathcal{C}}$. $\alpha = (x_1=0, x_2=1, x_3=0)$ since $R_{\overline{x}_1}$, R_{x_2}, and $R_{\overline{x}_3}$ are the only consistent reachability sets.

Thus, C has exactly one satisfying assignment.

LO6. Answer the following.

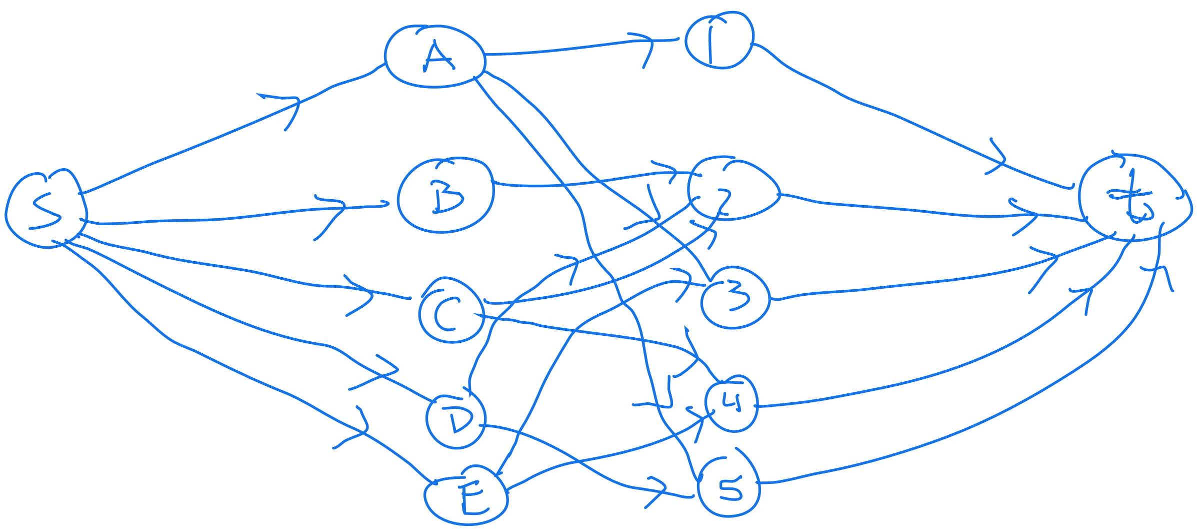
(a) Provide the definition of what it means to be a mapping reduction from problem A to problem B. (5 pts)

See Lecture Notes

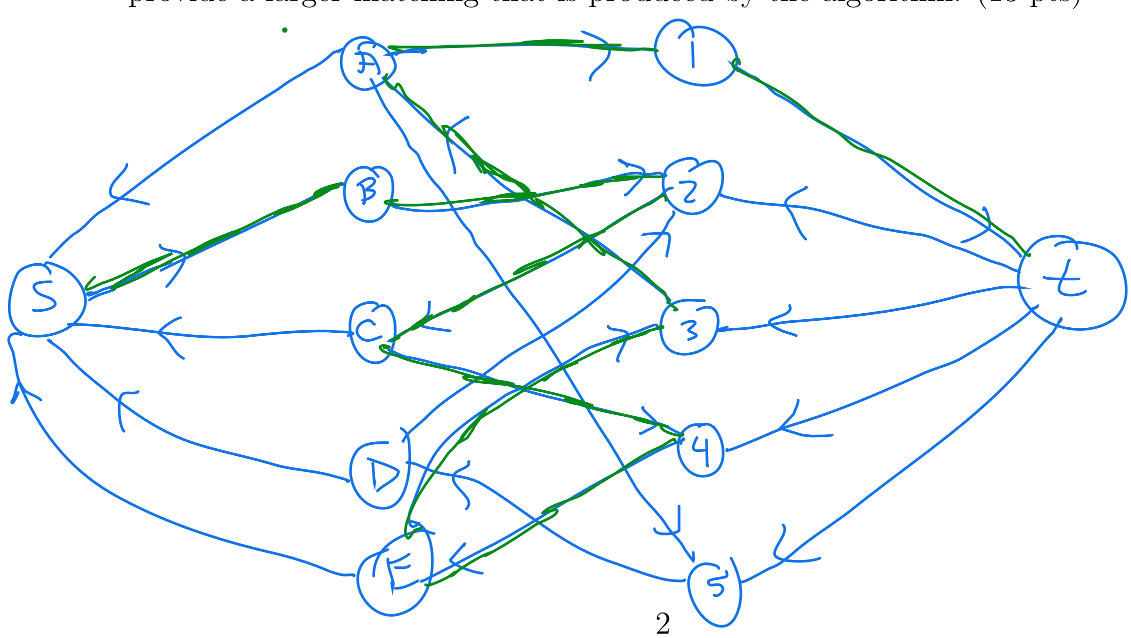
(b) An instance G of the `Max Bipartite Matching (MBM)` optimization problem has edge set

$$\{(A,1),(A,3),(A,5),(B,2),(C,2),(C,4),(D,2),(D,5),(E,3),(E,4)\}.$$

Draw $f(G)$, where f is the mapping reduction from `MBM` to `Max Flow` provided in lecture. Show work. (7 pts) Note: all capacities = 1



(c) Use part b and the Ford-Fulkerson Max-Flow Algorithm to decide if matching $M = \{(A,3),(C,2),(D,5),(E,4)\}$ is the maximum matching for G. If yes, explain why. If not, provide a larger matching that is produced by the algorithm. (13 pts)



Alternating Path: $P = s, B, 2, C, 4, E, 3, A, 1, t$

Max Matching: $M = \{(A,1), (B,2), (C,4), (E,3), (D,5)\}$

LO7. An instance of the `Quadratic Diophantine` decision problem is a triple (a, b, c) of nonnegative integers satisfying $a, b \le c$. The problem is to decide if there are nonnegative integers x and y for which $ax^2 + by = c$. We now establish that `Quadratic Diophantine` is an `NP` problem.

(a) For a given instance (a, b, c) of `Quadratic Diophantine`, describe a certificate in relation to (a, b, c). (7 pts)

(x, y) is a pair of numbers for which $x, y \le c$

(b) Provide a semi-formal verifier algorithm that takes as input i) an instance (a, b, c) of `QD`, ii) a certificate for (a, b, c) as defined in part a, and decides if the certificate is valid for (a, b, c). (7 pts)

Return $(ax^2 + by \le c)$.

(c) Problem `QD` has a single size parameter in relation to an instance (a, b, c). Describe this parameter. (6 pts)

$n = \lfloor \log c \rfloor + 1$.

(d) Use your answer to part c) to provide a bound on the number of steps required by the verifier from part b). Hint: multiplication is quadratic, while addition is linear. (5 pts)

The one-line verifier requires 4 $O(n^2)$ multiplications, one $O(n)$ addition and one $O(n)$ comparison for a total of $O(4n^2 + 2n) = O(n^2)$ steps which is a polynomial. Therfore, Quadratic Diophantine is an NP problem.

LO8. Answer the following.

(a) Let

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

be an instance of `3SAT`.

i. Compute $f(\mathcal{C}) = (S, t)$, where $f : \texttt{3SAT} \to \texttt{SS}$ is the mapping reduction from `3SAT` to `Subset Sum`. Provide the full table that shows the members of S, as well as the target value t.

				C_1	C_2	C_3	C_4
y_1	1	0	0	0	1	0	1
z_1	1	0	0	1	0	1	0
y_2	0	1	0	1	0	1	1
z_2	0	1	0	0	1	0	0
y_3	0	0	1	1	0	0	0
z_3	0	0	1	0	1	1	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	1	3	3	3	3

ii. Given that $\mathcal{C}$ is a positive instance of `3SAT` via satisfying assignment $\alpha = (x_1 = 0, x_2 = 1, x_3 = 0)$, provide the associated subset A of S that sums to t, verifying that (S, t) is a positive instance of `Subset Sum`. Hint: there is only one way of selecting the y's and z's to have a correct answer to this problem.

$$A = \{z_1, y_2, z_3, g_1, g_2, h_2, g_4\}$$

(b) The logical formula

$$\overline{(\overline{y}_1 \vee x_2)} \vee (y_2 \vee \overline{x}_1)$$

appears as part of the Tseytin transformation of some Boolean formula. Finish the transformation by using algebra to convert it to one or more `3SAT` clauses. (10 pts)

$$\overline{(\overline{y}_1 \vee x_2)} \vee (y_2 \vee \overline{x}_1) \iff (y_1 \wedge \overline{x}_2) \vee (y_2 \vee \overline{x}_1)$$

$$\iff (y_1 \vee y_2 \vee \overline{x}_1) \wedge (\overline{x}_2, y_2, \overline{x}_1)$$

$$\iff \{(y_1, y_2, \overline{x}_1), (\overline{x}_2, y_2, \overline{x}_1)\}$$

Advanced Problems

A1. Let A, B, and C be decision problems. Suppose we know that $A \leq_T B$ via an algorithm $\mathcal{A}$ that requires $O(n^5)$ steps and makes $O(n^3)$ queries to the B-oracle, where n is the size parameter for problem A. Also, we know that $B \leq_T C$ via an algorithm $\mathcal{B}$ that requires $O(m^7)$ steps and makes $O(m^2)$ queries to the C-oracle, where m is the size parameter for problem B.

(a) Explain how we may use the two algorithms to devise a third algorithm $\hat{\mathcal{A}}$ that establishes $A \leq_T C$. (15 pts)

On input x, Simulate $\mathcal{A}$ on input x. For each query $query_B(y)$ that occurs during the simulation answer the query by simulating $\mathcal{B}$ on input y.

(b) Provide good big-O upperbounds on both the running time of $\hat{\mathcal{A}}$ and the number of queries it makes to the C-oracle. Hint: the upper bounds should be in terms of n only. (20 pts)

$\mathcal{A}$ requires $O(n^5)$ steps. However, $O(n^3)$ of these steps are queries and each query now requires simulating $\mathcal{B}$ on an input of size $O(n^5)$. Moreover, since $\mathcal{B}$ requires $O(m^7)$ number of steps to execute, where $m = O(n^5)$, we have that $\hat{\mathcal{A}}$ requires

$$O(n^5) + O(n^3) \cdot O((n^5)^7) = O(n^{38}) \text{ steps}$$

and $O(n^3) \cdot O((n^5)^2) = O(n^{13})$ number of oracle queries to the C oracle.

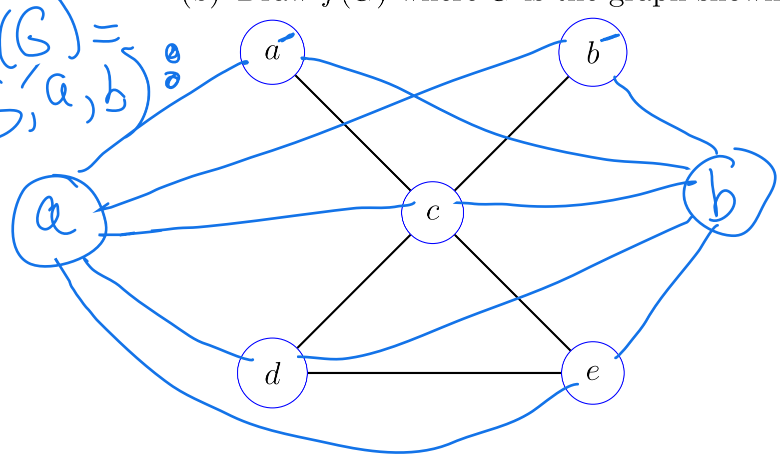
A2. You may have noticed that there are two versions of the `Hamilton Path` decision problem. An instance of the first version consists of a graph $G = (V, E)$ and two vertices $a, b \in V$. The problem is to decide if there is a Hamilton path in G that begins at vertex a and ends at vertex b. Call this version `Constrained Hamilton Path (CHP)`. An instance of the second version consists only of graph $G = (V, E)$ and the problem is to decide if there is a Hamilton path in G and it does not matter where the path begins or ends. Call this version `Free Hamilton Path (FHP)`.

(a) Provide a description for a *valid* mapping reduction $f : \texttt{FHP} \to \texttt{CHP}$ that maps an instance G of FHP to an instance $f(G) = (G', a, b)$ of CHP in such a way that G has a Hamilton path iff (G', a, b) has a Hamilton path from a to b. Prove that it is a valid reduction. (25 pts)

G' is equal to G, but with two added vertices a and b where each vertex (a and b) is joined to every vertex of G. Thus, if G has an HP, via some path P, then G' has an HP from a to b via the path aPb. Conversely, if G' has an HP from a to b of the form aPb, then P must be an HP for G. ◻

(b) Draw $f(G)$ where G is the graph shown below. (10 pts)

$f(G) = (G', a, b)$:



1 Unit 1 LO Problems (0 pts each)

LO1. Solve each of the following problems.

a. Use the Master Theorem to determine the growth of $T(n)$ if it satisfies the recurrence $T(n) = 4T(n/2) + n^2$.

b. Use the substitution method to prove that, if $T(n)$ satisfies

$$T(n) = T(n/2) + n/\log n,$$

Then $T(n) = \mathrm{O}(n/\log n)$. Hint: credit will *not* be awarded if the expression has been unjustly oversimplified.

LO2. Solve each of the following problems.

a. When performing the FFT algorithm for the purpose of evaluating the polynomial

$$p(x) = -8 + 5x + 4x^2 + 6x^3 - 2x^4 + 9x^5 + 3x^6 - 5x^7,$$

at the 8th roots of unity, the algorithm recursively calls FFT on two subproblem instances. What is the polynomial for each instance. In relation to these recursive calls, why is it important that the squares of the 8th roots of unity yield the 4th roots of unity?

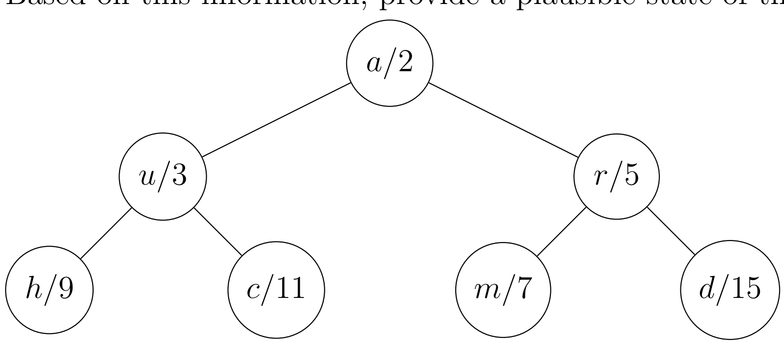
b. Consider Karatsuba's algorithm for multiplying two n-bit binary numbers x and y. Let x_L and x_R be the leftmost $\lceil n/2 \rceil$ and rightmost $\lfloor n/2 \rfloor$ bits of x respectively. Define y_L and y_R similarly. Let P_1 be the result of calling `multiply` on inputs x_L and y_L, P_2 be the result of calling `multiply` on inputs x_R and y_R, and P_3 the result of calling `multiply` on inputs $x_L + x_R$ and $y_L + y_R$. Then return the value $P_1 \times 2^{2\lfloor \frac{n}{2} \rfloor} + (P_3 - P_1 - P_2) \times 2^{\lfloor n/2 \rfloor} + P_2$. Demonstrate Karatsuba's algorithm on $x = 93$ and $y = 72$. Limit your demonstration to the root level of recursion. In other words, do not go deeper than level 0.

LO3. Solve each of the following problems.

a. The tree below shows the state of the binary min-heap at the beginning of some round of Dijkstra's algorithm, applied to some weighted *directed* graph G. Some of the edges of G include

$$(h, u, 2), (a, m, 4), (c, a, 3), (a, p, 2), (a, u, 5), (r, a, 2).$$

Based on this information, provide a plausible state of the heap at the end of the round.



b. An instance of the `Unit Task Scheduling` problem consists of the tasks shown in the table below.

Task	a	b	c	d	e	f	g	h	i	j	k
Deadline	4	8	6	6	4	2	3	2	8	7	1
Profit	50	80	30	30	70	30	70	50	80	20	60

State the greedy choice that is being made in each round of the algorithm and use it to obtain an optimal schedule. Show work.

LO4. Answer the following.

a. Provide the dynamic-programming recurrence for computing the distance $\mathrm{d}(u, v)$, from a single-source vertex u to a vertex v in a directed acyclic graph (DAG) $G = (V, E, c)$, where $c(x, y)$ gives the cost of edge $e = (x, y)$, for each $e \in E$. Hint: step *backward* from v, rather than forward from u.

b. Draw the vertices of the following DAG G in a linear left-to-right manner so that the vertices are topologically sorted, meaning, if (u, v) is an edge of G, then u appears to the left of v. The vertices of G are a-h, while the weighted edges of G are

$$(a, b, 9), (a, e, 5), (c, a, 3), (c, d, 1), (c, h, 2), (d, a, 4), (d, e, 1), (d, g, 3), (d, h, 4), (e, h, 6), (f, c, 4),$$

$$(f, d, 3), (f, e, 9), (g, b, 2), (g, e, 3), (h, b, 3).$$

c. Starting from left to right in topological order, use the recurrence to compute the distance from f to every vertex. Show the computations.