Note: Your TA probably will not cover all the problems. This is totally fine, the discussion worksheets are not designed to be finished in an hour. They are deliberately made long so they can serve as a resource you can use to practice, reinforce, and build upon concepts discussed in lecture, readings, and the homework.

1 Reducing Hamiltonian Cycle to Hamiltonian Path

(Also known as Rudrata Cycle and Rudrata Path in DPV.)

> **Hamiltonian Cycle:** Given a directed graph $G = (V, E)$, is there a cycle that visits every vertex exactly once?
>
> **Hamiltonian Path:** Given a directed graph $G = (V, E)$, is there a path that visits every vertex exactly once?

Given that Hamiltonian Cycle is **NP**-complete, prove that Hamiltonian Path is **NP**-complete.

(a) Show that Hamiltonian Path is in **NP**.

(b) Show that Hamiltonian Path is **NP**-hard by giving a polynomial-time reduction from Hamiltonian Cycle to Hamiltonian Path. Prove correctness in both directions.

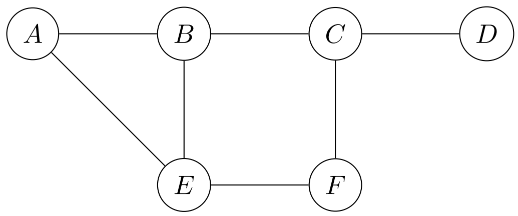
2 Vertex Cover to Set Cover

To help jog your memory, here are some definitions:

Vertex Cover: given an undirected unweighted graph $G = (V, E)$, a vertex cover C_V of G is a subset of vertices such that for every edge $e = (u, v) \in E$, at least one of u or v must be in the vertex cover C_V.

Set Cover: given a universe of elements U and a collection of sets $\mathcal{S} = \{S_1, \dots, S_m\}$, a set cover is any (sub)collection C_S whose union equals U.

In the *minimum vertex cover problem*, we are given an undirected unweighted graph $G = (V, E)$, and are asked to find the smallest vertex cover. For example, in the following graph, $\{A, E, C, D\}$ is a vertex cover, but not a minimum vertex cover. The minimum vertex covers are $\{B, E, C\}$ and $\{A, E, C\}$.



Then, recall in the *minimum set cover problem*, we are given a set U and a collection $\mathcal{S} = \{S_1, \dots, S_m\}$ of subsets of U, and are asked to find the smallest set cover. For example, given $U := \{a, b, c, d\}$, $S_1 := \{a, b, c\}$, $S_2 := \{b, c\}$, and $S_3 := \{c, d\}$, a solution is $C_S = \{S_1, S_3\}$.

(a) Give an efficient reduction from the minimum vertex cover problem to the minimum set cover problem, and prove its correctness.

(b) What does your reduction imply about the complexity of the minimum set cover problem? (You may assume minimum vertex cover is **NP**-hard.)

3 Maximum Coverage

In the maximum coverage problem, we have m subsets of the set $\{1, 2, \ldots, n\}$, denoted $S_1, S_2, \ldots S_m$. We are given an integer k, and we want to choose k sets whose union is as large as possible.

Give an efficient algorithm that finds k sets whose union has size at least $(1 - 1/e) \cdot OPT$, where OPT is the maximum number of elements in the union of any k sets. In other words, $OPT = \max_{i_1, i_2, \ldots i_k} |\cup_{j=1}^{k} S_{i_j}|$. Provide an algorithm description and justify the lower bound on the number of elements covered by your solution.

Hint: be greedy! For the proof, you may use the following property without proof: $(1 - 1/n)^n \leq 1/e$ *for all* $n \in \mathbb{Z}$.

