COSC 311: ALGORITHMS
MINI 10

Due Wednesday, December 11 in class
(Prof. Rager's section: please drop off at one of our offices)

The *Set Packing* problem is defined as follows:

Input:

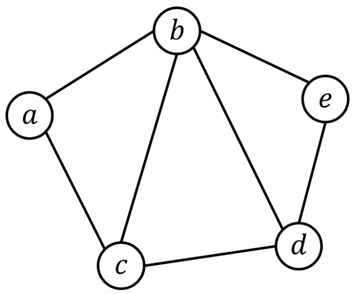
- A set U consisting of n elements
- m subsets of U, $S_1, \ldots, S_m$
- A positive integer k

Output: Is there a collection of at least k subsets such that no two subsets intersect?

The Set Packing problem can be seen as a generalization of Independent Set. In the Independent Set problem, we want to find a subset of nodes such that none of the nodes in our set are connected by an edge. Set Packing doesn't make any assumptions that the elements are organized into a graph structure or that conflicts between elements are explicitly encoded as edges. Nonetheless, both problems involve choosing a conflict-free set of items.

1. Give a polynomial-time reduction from Independent Set to Set Packing. Your answer should explain (1) given an instance of Independent Set, how do you turn it into an instance of Set Packing, and (2) how do you know that yes-instances of Independent Set map to yes-instances of Set Packing, and vice versa?

2. Show how your reduction from part (a) turns the following instance of Independent Set, with $k = 2$, into an instance of Set Packing:



3. Fill in the blanks to indicate what the reduction from Independent Set to Set Packing tells us about the relationship between the two problems:

If ____________________ cannot be solved in polynomial time, then neither can ____________________.

Explain why the above statement is true.