CS 473, Spring 2023
Homework 9 (due Apr 19 Wed 9pm)

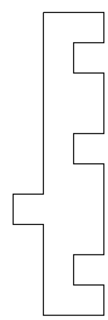
Instructions: As in Homework 1.

Problem 9.1: Prove that the following problem "JIGSAW-PUZZLE" is NP-complete:

> *Input*: a set P of n polygons where all vertices have integer coordinates and all edges are horizontal or vertical, and $M, N \leq 100n$.
>
> *Output*: yes iff there exists a placement of the n polygons so that they do not overlap and their union is exactly the $M \times N$ rectangle.

For simplicity, you may assume that we are only allowed to place each polygon by translation only, but *not rotation* (not even reflection).

(*Hint*: Reduce from HAMILTONIAN-PATH (a variant of the HAMILTONIAN-CYCLE), which you may assume is NP-complete. Pieces that look like the following may help. . .)



Problem 9.2: We are given a transportation network in the form of an undirected graph $G = (V, E)$. There are k factories each of which want to open two sites that are connected in G. In particular, factory i wants to open two given sites s_i and t_i in G (sites will be installed at the vertices of G) that are connected by some path P_i in G. Unfortunately, the roads in G are so narrow that they can support transportation service (say one truck) from only one factory at a time, i.e., we do not allow P_i and P_j to share vertices for any two distinct $i, j \in \{1, \dots, k\}$. Prove that given G and $s_1, t_1, \dots, s_k, t_k$, deciding whether there exists a feasible site installation (i.e., whether there exist paths $P_1, \dots, P_k$ satisfying the aforementioned constraints) is NP-complete.

(Hint: reduce from 3SAT. For each variable x_i, create a new (s_i, t_i) pair and connect them by 2 non-intersecting long paths—call this the "variable gadget". For each clause C_j, create a new (s_j, t_j) pair, and try to connect them with 3 paths, using vertices from the variable gadgets. . .)

Problem 9.3: We want to divide a set of *indivisible* items $S = \{1, 2, \dots, m\}$ among two agents 1 and 2. Each agent $i \in \{1, 2\}$ has a utility of u_{ij} (a positive integer) for item j. Given an allocation $X = \langle X_1, X_2 \rangle$ of S among the agents (i.e., S is the disjoint union of X_1 and X_2), we say that agent i *envies* agent j if and only if i prefers j's bundle to her own, i.e., $\sum_{k \in X_j} u_{jk} > \sum_{k \in X_i} u_{ik}$. We call an allocation *envy-free* if no agent envies another. Prove that

(a) deciding whether an envy-free allocation exists is NP-complete;

(b) given an integer r, deciding whether there exists an allocation $X = \langle X_1, X_2 \rangle$ such that $(\sum_{k \in X_1} u_{1k}) \cdot (\sum_{k \in X_2} u_{2k}) = r$ is NP-complete.

(Hint: Reduce from the Partition Problem.)