13th tutorial

MaxCut. In MaxCut, the task is to find an edge cut in an undirected graph of maximum size, i.e., a partition of vertices of G into two sets A, B such that the number of edges between A, B is as large as possible. Find a 1/2-approximation algorithm for MaxCut.

MaxSAT. We are given a logical formula in CNF which may not be satisfiable. We want to satisfy as many clauses as possible. Design a 1/2-approximation algorithm.

IndSet in Interval Graph. An interval graph G of a set of intervals $\{[x_1, y_1], \ldots, [x_n, y_n]\}$ has vertices $v_1, \ldots, v_n$ and has an edge $v_i v_j$ iff $|[x_i, y_i] \cap [x_j, y_j]| > 0$, that is, intervals i and j overlap non-trivially. Design a polynomial time algorithm which finds the maximum independent set in an interval graph.

TSP in 2^n instead of $n!$. Solving TSP by brute-force would enumerate all hamiltonian cycles, which would take roughly $\mathcal{O}(n!)$ time. Try to find a faster algorithm. Using dynamic programming, one can design an $\mathcal{O}(2^n \cdot n^k)$ algorithm for some constant k, which is still exponential, but much better than $\mathcal{O}(n!)$.