CMPSCI 611 FALL '17: HOMEWORK 5
DUE 8PM, MONDAY DECEMBER 11TH

- Homework may be completed in group of size with at most four students. You're not allowed to use material from the web or talk about the homework with anybody outside your collaboration group (aside from the lecturer or TA.)
- Solutions should be typed and uploaded as a pdf to `gradescope.com`.
- To get full marks, answers must be sufficiently detailed, supported with rigorous proofs (of both correctness and running time analysis). Faster algorithms will typically get more marks than slower algorithms.

Question 1. Let $[n] = \{1, \ldots, n\}$ and consider a collection of sets $\mathcal{B} = \{B_1, \ldots, B_m\}$ where each $B_i \subset [n]$. Let $b = \max_i |B_i|$.

(1) Design a polynomial-time approximation algorithm for the problem of choosing k sets $C_1, \ldots, C_k \in \mathcal{B}$ such that $| \cup_{i=1}^{k} C_i|$ is maximized. The algorithm should return a $1/(1 - (1 - 1/k)^k)$ approximation.

(2) Design a polynomial-time approximation algorithm for finding the smallest set $D \subseteq [n]$ such that $B \cap D \neq \emptyset$ for all $B \in \mathcal{B}$. The algorithm should return a b approximation.

Question 2. Let $G = (V, E)$ be an undirected graph in which each node has degree at most d. Design a polynomial time algorithm that finds an independent set whose size is at least $1/(d + 1)$ times that of the largest independent set.

Question 3. The input to the MIN-SQUARES problem is a set of positive integers $x_1, \ldots, x_n, r$ and L. We want to know if it is possible to partition $x_1, \ldots, x_n$ into r sets $S_1, S_2, \ldots, S_r$ such that

$$\sum_{i=1}^{r} \left(\sum_{x \in S_i} x \right)^2 \leq L \ .$$

Hint: Note that $\sum_{i=1}^{r} \left(\sum_{x \in S_i} x\right)^2 \geq r \left(\sum_{i=1}^{n} x_i/r\right)^2$.

(1) Prove that MIN-SQUARES is NP-Complete via a reduction from SUBSETSUM.

(2) Design (and analyze) a factor 4 approximation algorithm for minimizing $\sum_{i=1}^{r} \left(\sum_{x \in S_i} x\right)^2$.

Question 4. A set of vertices V' in an undirected graph $G = (V, E)$ is called a *dominating set* if every vertex v of G is either in V' or there is an edge $(v, u) \in E$ such that $u \in V'$. Prove that the problem of determining if a graph G has a dominating set of size k is NP-complete. [*Hint: Consider a reduction from vertex cover.*]

Question 5. The winter of 1931 was so bad in Amherst that UMass was closed for almost 6 weeks. Fortunately, the algorithms class that year had prepared for this possibility:

- The lecturer had nominated t of the n students in the class as assistant lecturers and provided each with a copy of the lecture materials.
- Instead of coming to school for the lectures, each student only had to travel to the house of the closest assistant lecturer.

To ensure that each student did not have to walk too far, the lecturer had taken great care when nominating the students who would serve as assistant lecturers. In particular, suppose that student

i and student j live d_{ij} miles from each other (you may assume that for all i, j, k, $d_{ij} + d_{jk} \geq d_{ik}$). The lecturer wanted to nominate a subset C of t students with the goal of minimizing:

$$\text{max-dist}(C) = \max_{i \in [n]} \min_{j \in C} d_{ij}$$

(1) Design a polynomial time reduction from Dominating Set to the decision problem "Does there exist a subset C of size t such that $\text{max-dist}(C) < 2$?" This establishes that the problem is NP-Hard.

(2) Consider the following approximation algorithm for the picking C:
  - (a) Let $C = \{1\}$
  - (b) For $r = 2$ to t: $C \leftarrow C \cup \{i\}$ where i maximizes $\min_{j \in C} d_{ij}$
  - (c) Return C

  What is the approximation ratio of the algorithm? Prove your answer.