

CSCI 570 - Spring 2018 - HW 9

1. At a dinner party, there are n families a_1, a_2, \dots, a_n and m tables b_1, b_2, \dots, b_m . The i^{th} family a_i has g_i members and the j^{th} table b_j has h_j seats. Everyone is interested in making new friends and the dinner party planner wants to seat people such that no two members of the same family are seated in the same table. Design an algorithm that decides if there exists a seating assignment such that everyone is seated and no two members of the same family are seated at the same table.
2. There is a precious diamond that is on display in a museum at m disjoint time intervals. There are n security guards who can be deployed to protect the precious diamond. Each guard has a list of intervals for which he/she is available to be deployed. Each guard can be deployed to at most A time slots and has to be deployed to at least B time slots. Design an algorithm that decides if there is a deployment of guards to intervals such that each interval has either exactly one or exactly two guards deployed.
3. Solve Kleinberg and Tardos, Chapter 7, Exercise 28.
4. The computer science department course structure is represented as a directed acyclic graph $G = (V, E)$ where the vertices correspond to courses and a directed edge $(u, v) \in E$ exists if and only if the course u is a prerequisite of the course v . By taking a course $w \in V$, you gain a benefit of b_w which could be a positive or negative number. Design an algorithm that picks a subset $A \subset V$ of courses to take such that the total benefit $\sum_{w \in A} b_w$ is maximized. Remember that if $v \in A$ and $(u, v) \in E$, then u has to be in A . That is, to take a course, you have to take all its prerequisites. The running time should be polynomial in $|V|$.