
Efficient Algorithms and Datastructures I

Question 1 (10 Points)

The Mathematics and Computer Science department has n faculty members f_1, f_2, \dots, f_n who will offer n courses c_1, c_2, \dots, c_n in the coming semester and each faculty member will teach exactly one course. Each faculty member chooses two courses he (or she) would like to teach, and ranks them according to his (or her) preference (rank 1 indicates higher preference and rank 2 indicates lower preference).

- (a) We say that a course assignment is a *feasible* assignment if every faculty member teaches a course within his (or her) preference list. How would you efficiently determine whether the department can find a feasible assignment?
- (b) We say that a feasible assignment is an *optimal assignment* if it maximizes the number of faculty members assigned to their most preferred course. Suggest an efficient algorithm for determining an optimal assignment and analyze its complexity.

Question 2 (10 Points)

The dean needs to assign a group of n faculty members to be chairs on n committees. Each faculty member proposes, in decreasing order of preference, a list of three committees that he or she would like to chair. We want to determine whether there exists a *satisfiable assignment* (one that assigns the faculty to the committees so that each faculty member obtains a job on his or her list). If some satisfiable assignment is possible, we want to find the assignment that maximizes the number of faculty with their most preferred committee chair, and further, among such assignments, the assignment that maximizes the number of faculty with their second most preferred committee chair. Show how to solve this problem by solving a single assignment problem.