Due: Tuesday July 24th at 6:00PM in class, or Thursday July 26th at 6:00PM electronically
Worth: 4%.
For each question you have to clearly write your algorithm in English and prove that it finds the optimal answer. In any of your answers you can use any algorithm we discussed in class without proving it solves the problem we discussed in class optimally. If we discussed the runtime of the algorithm you can also use that without reproving it. The same goes for any Lemma, Theorem or Fact we discuseed in class.
This assignment is about Maximum Flow and Linear Programming. For the first question there is a solution that "reduces" the problem to Max. Flow, i.e. given an input to the problem shows how to construct a flow network G such that one can construct the solution to the problem from the maximum flow of G. If you think you have another way to solve this problem (efficiently) you are welcome to write that but I highly doubt such a solution exists (or is easy to find.) Make sure you describe how the flow network should be constructed from the input to the problem and prove that the max-flow in the network will give you the optimal solution to the original problem.
For the second question you are *required* to formulate the problem as an LP. That is given an input to the problem construct a linear program the solution to which would be (or give) the solution to the problem. There are polynomial time algorithms to solve this question directly but *we are not asking for those!* You will only get the marks if you formulate the problem as an LP.

Question #1: The hiring problem(10pt)

In class we solved the following problem by formulating it as a network flow problem: "Given n applicants and m positions and a list of positions each applicant is compatible with how can we match the maximum number of applicants to position while making sure that no applicant is matched to more than one positions and no position is matched to more than one applicant." If you don't remember the solution you can take a look at Section 26.3 of CLRS. Here we will solve a generalization of that problem.

Part a (4pt):

We have n applicants who have applied for jobs in our company. There are m roles available at the company for which we are hiring applicants but for some of these roles we can hire more than one applicant. In particular we are given numbers $a_1, a_2, \ldots, a_m$ and we can hire up to a_j people for role j. Like before we also have sets $C_1, C_2, \ldots, C_n$ where each C_i is a subset of $\{1, \ldots, m\}$ and applicant i is compatible with roles in the set C_i, i.e. we can hire applicant i for any of the roles in the set C_i.
Given n, m, $C_1, \ldots, C_n$ and $a_1, \ldots, a_m$. Find an assignment of applicants to roles such that:

1. No applicant is assigned to more than one role. Some applicants can be assigned to no role in which case we will not hire them,
2. There are no more than a_j applicants assigned to role j,
3. Applicant i can only be assigned to roles in C_i,
4. The number of applicants hired, i.e. the number of applicants assigned to roles is as big as possible.

To get the full marks the number of vertices and edges in your flow network should not *depend on a_i's.*

Examples: If $n = 4, m = 2$, $C_1 = \{1, 2\}, C_2 = \{2\}, C_3 = \{1\}, C_4 = \{1, 2\}$, $a_1 = 2, a_2 = 1$. Then the answer is 3. We can hire applicants 1 and 3 for role 1 and candidate 2 for role 2.
If $n = 4, m = 2$, $C_1 = \{1, 2\}, C_2 = \{2\}, C_3 = \{2\}, C_4 = \{2\}$, $a_1 = 2, a_2 = 1$. Then the answer is 2. We can hire applicants 1 for role 1 and candidate 2 for role 2.
If $n = 4, m = 4$, $C_1 = \{1, 2\}, C_2 = \{1\}, C_3 = \{2\}, C_4 = \{3, 4\}$, $a_1 = 1, a_2 = 1, a_3 = 1, a_4 = 2$. Then the answer is 3. We can hire applicant 2 for role 1, applicant 3 for role 2 and applicant 4 for role 4.

Part b (6pt):

We want to solve the question in part a but there is a further restriction. Now we also have d possible field of expertise. Each applicant comes with a number e_i between 1 and d meaning that the field of expertise of applicant i is e_i. For any of our role we can not hire two people with the same field of expertise.
Given n, m, d, $C_1, \ldots, C_n$, $e_1, \ldots, e_n$ and $a_1, \ldots, a_m$. Find an assignment of applicants to roles such that:

1. No applicant is assigned to more than one role. Some applicants can be assigned to no role in which case we will not hire them,
2. There are no more than a_j applicants assigned to role j,
3. If applicants i and i' are both assigned to role j then $e_i \neq e_{i'}$,
4. Applicant i can only be assigned to roles in C_i,
5. The number of applicants hired, i.e. the number of applicants assigned to roles is as big as possible.

Examples: If $n = 4, m = 2, d = 2$, $C_1 = \{1, 2\}, C_2 = \{2\}, C_3 = \{1\}, C_4 = \{1, 2\}$, $e_1 = 1, e_2 = 1, e_3 = 1, e_4 = 2$, $a_1 = 2, a_2 = 1$. Then the answer is 3. We can hire applicants 1 and 4 for role 1 and candidate 2 for role 2.
If $n = 4, m = 2, d = 2$, $C_1 = \{1, 2\}, C_2 = \{2\}, C_3 = \{1\}, C_4 = \{1, 2\}$, $e_1 = 1, e_2 = 2, e_3 = 1, e_4 = 1$, $a_1 = 2, a_2 = 1$. Then the answer is 2. We can hire applicant 1 for role 1 and candidate 2 for role 2. Although $a_1 = 2$ we can't hire two applicants for this role because all the compatible applicants have the same field of expertise.
If $n = 4, m = 3, d = 3$, $C_1 = \{1, 2\}, C_2 = \{2\}, C_3 = \{1, 2, 3\}, C_4 = \{3\}$, $e_1 = 1, e_2 = 2, e_3 = 3, e_4 = 3$, $a_1 = 1, a_2 = 1, a_3 = 2$. Then the answer is 3. We can hire applicant 1 for role 1, applicant 2 for role 2 and applicant 4 for role 3.

Question #2: Numbers on a path again! (6pt)

Formulate the following problem as a linear program. (This is pretty much problem 1 from assignment 1 but the greedy solution for that problem will not help you in formulating it as an LP.)
Given n real numbers $a_1, \ldots, a_n \in \mathbb{R}$. Find n real numbers $b_1, \ldots, b_n \in \mathbb{R}$ with the requirement that the b_i's form a non-decreasing sequence, i.e. $b_1 \leq b_2 \leq \cdots \leq b_n$ and they have the minimum *deviation*. The deviation of b_i's is the maximum distance between an a_i and its corresponding b_i, i.e.,

$$\text{deviation} = \max(|a_1 - b_1|, |a_2 - b_2|, \ldots, |a_n - b_n|).$$

Hints:

1. First try to formulate this problem as a linear program if deviation was defined as $\max(a_1 - b_1, \ldots, a_n - b_n)$.
2. Remember that the objective value and constraints in a linear program have to be *linear*, in particular you can't use the max function in them. You need a trick. How about adding an extra variable?
3. How can you bring the absolute values into the picture now?

Examples: If $n = 2$, $l = 10$, $a_1 = 4$, $a_2 = 2$ then the minimum deviation is 1. An optimal answer is $b_1 = 3$, $b_2 = 3$.