Note: There can be multiple correct solutions to some of the problems. We (TAs) provide one of such solutions. If you think you have nice and elegant solution to any of the problems, send it to us. We will put it here.

1. You are given a bipartite graph G on $2n$ vertices and a matching M of size n. That is every vertex is the end-point of one edge in the matching.

   An alternating cycle is a cycle that alternately consists of edges which are in the matching and those not in the matching.

   (a) * Design a polynomial time algorithm to find an alternating cycle in G if one exists. Describe clearly the main ideas of your algorithm in English. Please do not write code. *(10 marks)*

   SOLUTION:

   (7 marks) Given a matching edge $e_M \equiv (u, v)$ do the following.

   1: Remove e_M from the graph G temporarily
   2: Apply the algorithm done in class to find an augmenting path between u and v in the modified graph
   3: If an AP is found output the cycle made of e_M and the AP

   Alternatively, you could merge the steps 1, 2 & 3 and directly find the cycle using the modified BFS done in class staring from either u or v, considering e_M first.

   (3 marks) Repeat the above procedure for each matching edge in M.

   (b) Using the algorithm from exercise (*) above design an algorithm to test if this is the only matching of this size. Prove that your algorithm is correct. You will get no credit if your algorithm does not use the previous exercise. *(10 marks)*

   SOLUTION:

   (5 marks) Run the algorithm (*) to find an alternating cycle in G. Output YES (i.e. M is the only matching of size n) if no alternating cycle can be found. Otherwise ouput NO.

   (5 marks) **Proof of correctness**

   $\Rightarrow$ G has an alternating cycle. Then there is another matching of same size. The new matching can be found by switching matching and non-matching edges in the AC. Since all edges connecting the AC to other vertices in G are non-matched, switching does in fact create a valid matching. Since there are equal number of matching and non-matching edges in AC, switching does not change the size of the matching.

   $\Leftarrow$ There is another matching M' of same size. Then G has an alternating cycle. The AC is nothing but $M \oplus M'$. This is because, in class we proved that $M \oplus M'$ contains either "odd or even length alternating paths" or "even length alternating cycles".

   i. Since both M and M' are maximal matching of size n the possibility of "even length paths" is ruled out because otherwise there will be an unmatched vertex in one which is not possible.
   ii. The possibility of "odd length paths" is ruled out because, in that case we could find an AP w.r.t. one indicating that there is a matching of greater size which is not possible.

   The only possibility left is "even length alternating cycles".

   ---

2. Solve the following linear program using the primal-dual method (As we did for msts and shortest paths). **Solving it any other way will not fetch you any marks.** Show each of your steps clearly, so we know you have used this method. *(20 marks)*

$$
\begin{aligned}
\min\ & 4x + 9y + z \\
\text{s. t. } & x + 2y + z = 5 \\
& 3y - 2z = -3 \\
& x,\ y,\ z \geq 0
\end{aligned}
$$

SOLUTION:

(a) *(3 marks)* In order to carry out the primal-dual algorithm, we must first write down the dual. For the given problem, the simplest dual is

$$\begin{aligned} \text{max } 5a - 3b & \\ \text{s. t. } a & \leq 4 \\ 2a + 3b & \leq 9 \\ a - 2b & \leq 1 \end{aligned}$$

Note that the dual variables are free, i.e. $a,\ b \lesseqgtr 0$.

Some students have converted the primal equalities to inequalities and then obtained the dual. This is OK, as long as you realise that then the dual variables must be ≥ 0. Also, this increases the number of dual variables.

(b) *(3 marks)* To start off with the primal-dual algorithm, we need an initial feasible solution for the dual. The simplest feasible solution here is $a = b = 0$.

(c) *(4 marks)* Next, we look for a way to increase the dual cost. Looking at the objective function we see that a has a positive coefficient, while b has a negative coefficient. Hence, we increase a while keeping $b = 0$ until some constrataint becomes tight. This occurs at $a = 1$, when the third inequality becomes tight.

(d) *(4 marks)* Our current dual solution is $a = 1,\ b = 0$ and the cost is $5a - 3b = 2$. In order to improve this, we write the auxiliary dual:

$$\begin{aligned} \text{max } 5a' - 3b' & \\ \text{s. t. } a' - 2b' & \leq 0 \end{aligned}$$

We know (from the last question in the second quiz!) that our solution to the dual is optimal iff the maximum value of this auxiliary dual is 0. However we see that the cost of this auxiliary dual can be increased by increasing a' and b'. Also, the inequality is kept tight if $a' = 2b'$. Hence, a possible solution to this is $(2, 1)$.

To get a solution to the dual from this, we try to find the largest value of λ so that $a = a + \lambda a'$, $b = b + \lambda b'$ is still feasible. We find that this occurs at $\lambda = 1$, and the second constraint is tight. Hence we get $a = 3, b = 1$. Our cost is $5a - 3b = 15$.

(e) *(4 marks for showing optimality, 2 marks for primal variables)* In order to verify that this is the optimal solution, we can write the auxiliary dual again, and see if we can increase the objective function. Another way is to use complementary slackness. The second method has the advantage that we can also obtain the primal variables.

We know that solutions x, y, z and a, b to the primal and dual are optimal iff (a) they are feasible, and (b) whenever a primal variable is greater than zero, the corresponding dual constraint is tight, and vice versa. Here we see that the first dual constraint is not tight, hence the first primal variable has to be zero. Using this, we try to find a feasible solution to the primal. Solving

$$\begin{aligned} 2y + z & = 5 \\ 3y - 2z & = -3 \\ y, z & \geq 0 \end{aligned}$$

we get $y = 1,\ z = 3$. By complementary slackness, this is then an optimal solution. Hence our optimal value is 15, and this is obtained for $x = 0,\ y = 1,\ z = 3$. Note that the objective functions of the primal and dual coincide, as they must.

---

3. You are given as input a directed graph with positive integer weights on edges. We want to output a directed cycle of minimum weight spanning all vertices of this graph. Formulate this as an integer linear program.

You get full credit if your formulation is correct and it uses a polynomial number of constraints. Half credit for any correct formulation.

SOLUTION:

We give the solution for exponential number of constraints. We (the TAs) are not sure of a way to do this with a polynomial number of constraints.

Note that an ILP is a linear program with the added constraint that the variables are constrained to be integers.

For each directed edge $(u, v) \in E$, we have a variable x_{uv}.

(a) *(3 marks)* Objective function and constraints on x_{uv}

$$\begin{aligned} & \text{min } \textstyle\sum_{(u,v)\in E} w_{uv} x_{uv} \\ & x_{uv} \quad \geq 0 \\ & x_{uv} \quad \text{is an integer} \end{aligned}$$

(b) *(3 marks)* Connectivity constraints

$$\forall S \subset V \quad \textstyle\sum_{u \in S, v \notin S, (u,v)\in E} x_{uv} \geq 1$$

Note that it is sufficient to say that one edge must leave for each subset S, since we will consider both S and $V - S$ in different constraints. Note also that this gives a exponential number of constraints. Without these, we could get multiple disconnected cycles, as shown in figure 1.

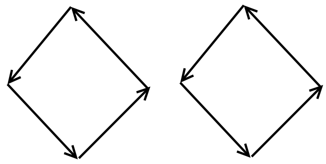

Figure 1: A disconnected graph

(c) *(4 marks)* To ensure we obtain a cycle

$$\begin{array}{lll} \forall v \in V & \sum_{u \in V} x_{uv} & = 1 \\ \forall v \in V & \sum_{w \in V} x_{vw} & = 1 \end{array}$$

The constraints say that each vertex has exactly one edge entering and one leaving it. Another acceptable set of constraints is

$$\begin{array}{lll} \forall v \in V & \sum_{u \in V} x_{uv} - \sum_{w \in V} x_{vw} & = 0 \\ \forall v \in V & \sum_{w \in V} x_{vw} & = 1 \end{array}$$

However, the following set of constraints are not sufficient:

$$\begin{array}{lll} \forall v \in V & \sum_{u \in V} x_{uv} - \sum_{w \in V} x_{vw} & = 0 \\ \forall v \in V & \sum_{w \in V} x_{vw} & \geq 1 \end{array}$$

as we may then have a vertex being in two cycles, as shown in the figure.

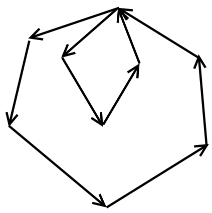

Figure 2: A tour rather than a cycle