You are allowed to discuss with others but are not allowed to use any references other than the course notes. Please list your collaborators for each question. You must write your own solutions.

There are totally 70 marks including the bonus programming problem. The full mark is 50. This homework is counted 5% of the course grade.

---

1. **Written Problem: Facility Scheduling** (12 marks)

   There are m facilities and n people. Each person requests to use a subset of facilities. Each facility can only be used by one person in one day. Suppose each person requests to use at most d facilities, and each facility is requested by at most d people. Model this as a graph problem and prove that there is a schedule (which person to use which facility on which day) that uses at most d days so that every person can use all the facilities that he/she requested. Design an efficient algorithm to output such a schedule. You will get full marks if the time complexity is polynomial and the proofs are correct.

   Table 1: Input: Facility Requests

	Adam	Mary	John	Peter
Soccer	✓		✓	
Basketball	✓	✓		
Tennis			✓	✓

   Table 2: Output: Final Schedule

	Day 1	Day 2
Soccer	Adam	John
Basketball	Mary	Adam
Tennis	John	Peter

2. **Written Problem: Hamiltonian Cycle** (7 marks)

   Prove that the Hamiltonian Cycle problem is NP-complete even when restricted to undirected bipartite graphs.

3. **Written Problem: Degree Bounded Spanning Tree** (7 marks)

   Prove that the following problem is NP-complete.

   Input: An undirected graph $G = (V, E)$ and an positive integer k.

   Output: Does there exist a spanning tree $T \subseteq E$ with maximum degree at most k?

4. **Written Problem: Minimum Covering Set** (12 marks)

   Prove that the following problem is NP-complete.

   Input: An undirected graph $G = (V, E)$ and a positive integer k.

   Output: Does there exist a covering set with at most k vertices? Recall from HW4 that a subset of vertices $S \subseteq V$ is a covering set if for every vertex $u \in V - S$ there exists a vertex $v \in S$ such that $uv \in E$.

5. **Written Problem: Directed Acyclic Subgraph** (12 marks)

   Prove that the following problem is NP-complete.

   Input: A directed graph $G = (V, E)$ and a positive integer k.

   Output: Does there exist a subset $F \subseteq E$ with at most k edges such that $G - F$ is directed acyclic?

6. **(Bonus) Programming Problem: Flu Detection Machine** (20 marks)

   There are a lot of flights flying back and forth between country A and country B. However, as it is a high season of flu, both governments would like to cooperate to protect their countries from wide range of infections.

   The governments plan to install powerful flu detection machines in some airports, such that all passengers are scanned by the machine at least once either at the departure airport or the arrival airport.

   Since that machine is quite expensive, the governments would like to minimize the number of machines they must install in the airports.

   Given all the flight information between the two countries, your task is to write a program to find out which airports to install the machines.

   INPUT: The first line consists of three integers N, M, F, representing the number of airports in country A, country B, and the number of flights respectively. Airports in country A are numbered by $1, 2, \ldots, N$ and airports in country B are number by $1, 2, \ldots, M$. Then, each of the following F lines consists of two integers a and b ($1 \leq a \leq N$, $1 \leq b \leq M$), meaning that there is a flight between the airport a in country A and airport b in country B.

   It is guaranteed that $1 \leq N, M \leq 200$ and $0 \leq F \leq N \times M$.

   OUTPUT: The first line consists of a single integer K, the minimum total number of machines that must be installed. Then K lines follows, where each line consists of a character x and an integer y separated by a space, meaning to install a machine in airport y in country x. If there are multiple solutions, you can output any one of them.

   SAMPLE INPUT:
   ```
   3 3 4
   1 2
   2 1
   3 3
   2 3
   ```

   SAMPLE OUTPUT:
   ```
   3
   A 2
   B 2
   B 3
   ```

   Explanation: By installing a machine in the airport 2 in country A, passengers in the second flight are scanned. Similarly. the machine in airport 2 in country B scans the first flight and the machine in airport 3 in country B scans the third and the fourth flight. Hence all possible passengers are scanned.