Alice wants to throw a party and is deciding who to invite. She has n people to choose from, and she has created a list consisting of pairs of people who know each other. She wants to invite as many people as possible, subject to the following two constraints.

- Each person should have at least five other people whom they know.
- Each person should also have at least five other people whom they do not know.

Formally, you are given the list of n people and the list of all pairs of people who *mutually* know each other; that is, you may assume that if X knows Y, Y also knows X. To make things simpler, we can model the problem as a graph. Each person is a vertex, and an *undirected* edge connects two people X, Y if and only if X and Y mutually know each other. Therefore, the input can be represented by a graph $G = (V, E)$ and an adjacency matrix $A[1..n, 1..n]$.

(a) Describe an $O(n^3)$ algorithm to return a subset of maximum cardinality of people satisfying the two constraints.

(b) Argue that your algorithm produces an optimal subset.

(c) Prove that the optimal subset is *unique*.

Hint. *Union-closed families have a unique "largest element".*

Rubric.

- You should state clearly why your algorithm is correct and runs in the allocated time complexity (or faster!).
- This task will form part of the portfolio.
- Ensure that your argument is clear and keep reworking your solutions until your lab demonstrator is happy with your work.