Signe Tudent s101010

Exercise 1.1

We construct an undirected graph where the vertices are persons, and the edges are friendships.

Exercise 1.2

Algorithm We construct the graph G for the party and determine whether it is bipartite. If it is, we output "yes, there is a partition;" if not, we output "no, there is no partition."

Correctness By definition, it is possible to partition the persons into two teams, T_1 and T_2, if and only if the corresponding graph is bipartite. Hence, the algorithm is correct.

Analysis We use $O(x+y)$ time to construct the graph and $O(x+y)$ time to determine if it is bipartite. In total, we use $O(x+y)$ time.

Exercise 1.3

Algorithm We construct the graph G for the party, run a DFS on G from p_0, and count the number of visited vertices. If the number of visited is at least k, we output "yes, we can reach at least k persons," if not, we output "no, we cannot reach k persons."

Correctness A person p' is reachable from a person p if and only if there is a path between p to p' in G. A DFS from p_0 finds all vertices connected to p_0. Hence, counting these determines the number of vertices reachable from p_0. Hence, the algorithm is correct.

Analysis We use $O(x+y)$ time to construct the graph and $O(x+y)$ time to do the DFS. In total, we use $O(x+y)$ time.

Exercise 1.4

We construct the graph G for the party and assign a weight on each edge that is the strength of the corresponding friendship. We now binary search for the smallest weight s in the range $[1, 10x^2]$ such that the graph G_s consisting of all edges of weight no more than s is connected. Each step of the binary search proceeds as follows:

1. Construct graph G_s.
2. Run DFS on G_s from p_0.
3. If G_s is connected we lower s and otherwise we increase s.

We output the smallest s determined by the binary search.

Correctness The graph G_s consists of the edges with weight $\leq s$ from G. Thus, G_s corresponds to the party with all friendships of strength $\leq s$. The algorithm is correct since we output the smallest s such G_s is connected.

Analysis Each binary search step uses $O(x+y)$ time. We binary search over a range of size $10x^2$, and thus use $O(\log(10x^2)) = O(\log x)$ binary search steps. In total, we use $O((x+y)\log x)$ time.