1. **specifications**

In a group of people, it is to be expected that some of them may not want to work with each other. Assuming that each person has at most d other people that they don't want to work with, divide the people into $d+1$ groups so that everyone is in exactly one group and no one is in a group with someone they don't want to work with.

Input: n people, for each person, up to d other people they don't want to work with

Output: membership for each of the d+1 groups, or labeling of the group each person belongs to

Legal solution: each person is in exactly one group, no one is in a group with someone they don't want to work with

2. **examples**
3. **targets**
4. **tactics**
5. **approaches**

process input – for each person, assign them to a group

produce output – who's the next person in this group? (for each group, assign its members)

~~narrow the search space –~~

6. **main steps**

for each person

   go through the groups, assign the person to the first group that doesn't have someone they don't want to work with

7. **exit condition**

when everyone has been assigned to a group

8. **setup**

9. **wrapup**

10. **special cases**

- is the don't want to work together mutual?

- can we have fewer than d+1 groups?

11.algorithm

12.termination
   a) **measure of progress**
   b) **making progress**
   c) **reaching the end**

13.correctness
   a) **loop invariant**

for the first k people assigned to a group, no one has been assigned to more than one group and no one is in a group with someone they don't want work with

   b) **establish the loop invariant**
   c) **maintain the loop invariant**
   d) **final answer**

14.implementation

15.time and space