Problem [Lovász toggle] Let $G = (V, E)$ be an undirected graph. Assume every vertex of $G$ has degree $\leq x + y + 1$ ($x, y$ are given integers). We wish to color the vertices red or blue (each vertex gets exactly one color) such that each red vertex has at most $x$ red neighbors and each blue vertex has at most $y$ blue neighbors. (Note that this is not a legal coloring in the sense of the definition of the chromatic number.) Show that this is always possible, using the following algorithm (given here in pseudocode).

procedure Lovász-toggle

1. Initialize by coloring each vertex arbitrarily
2. Call a vertex “bad” if it has more than the permitted number of neighbors of its own color
3. BAD := set of bad vertices
4. while BAD $\neq \emptyset$
5. pick a bad vertex
6. switch its color
7. update BAD
8. end(while)

(a) Prove that this algorithm will terminate in a finite number of steps. (Give a very simple and convincing argument, no more than 5 or 6 lines.) Give an upper bound on the number of cycles of the while loop in terms of the basic parameters $|V|, |E|$. Hint. Call the graph with a coloring a “configuration.” With each configuration, associate an integer (the “potential”) in such a way that each round of the Lovász-toggle reduces the potential. This will give a bound on the number of rounds. Note that “the number of bad vertices” is NOT an appropriate potential function: it can increase.

(b) Show that statement (a) becomes false if the degree bound is increased to $x + y + 2$. Construct graphs where each vertex has degree $\leq x + y + 2$ and where

- the algorithm never terminates, regardless of the initial coloring and the choice of bad vertex to be switched on line 5;
- for some initial colorings and some choices of the bad vertex to be switched, the algorithm will terminate, for others it will not.