Advanced Algorithms	HW22, 9.9	Michael Lewellen

(9.9) In the maximum cut problem we are given an undirected graph, $G = \{V, E\}$ with a weight $w(e)$ on each edge.

We wish to separate the vertices into two sets S and $V-S$ so that the total weight of the edges between the two sets is as large as possible.

For each $S \subseteq V$, define $w(S)$ to be the sum of all w_{uv} over all edges $\{u,v\}$ such that $|S \cap \{u,v\}| = 1$ maximize $w(S)$ over all subsets of V.

Consider this Algorithm:

start with any $S \subseteq V$
While there is a subset $S' \subseteq V$ such that
$|(S'-S) \cup (S-S')| = 1$ and $w(S') > w(S)$ do:
set $S = S'$

(a) show this is an approximation algorithm for Max Cut with ratio of 2.

while the algorithm is worded stupidly, it means find a nearby set (local search) and see if it gets better. Repeat until stuck

– First does the algorithm terminate? There is an m = max # of edges –

Each step of the algorithm increases the maximum by at least one for $w(S') > w(S)$, so it will terminate in a maximum of m steps. it can not increase past m.

I really can't speculate what is meant by ratio of 2 but if our algorithm keeps exploring locally it will produce at least a local maximum which is an approximation of MAX CUT

(b) The approximation should be polynomial since we are only expanding locally relative to any set we arrive at a local maximum.

Since the algorithm says for "any S". We repeat for other sets in the graph we can solve for a maximum approximation in polynomial time –