Assignment 6 Problem 1

Since the graph is finite, and . The only possible that can let Runner gets infinitely larger than the Coyote is there exists a cycle such that there is a path from s to that cycle, and each time passing the cycle, the value of getting greater, which means that the product of all in that cycle is less than the product of all in that cycle. By go through that cycle again and again, will getting greater and greater which will go to infinity.

Naming this kind of cycle “special cycle”.

To find the cycle, applying DFS on vertex s. When we meet some vertex that is discovered already (which means we have found a cycle!!), then we back track it, by track its parents, and track parents’ parents, and so on. Until we reach the last occurrence of such vertex. During the back track process, also calculate the value of for all edges that we tracking through (these edges are in the cycle). After going through such cycle, if we find the value of is greater than 1, then such cycle is a ‘special cycle’, and thus by going through this cycle again and again, then Runner gets infinitely larger than the Coyote.

Pseudo code:

DFS(v)

mark(v) = discovered

for u AdjacencyList(v) do

if u is undiscovered then

DFS(u); parent(u) = v;

else

back track the cycle through parent array

in the same time calculate for all edges in the cycle

if return answer exist ‘special cycle’

fi

Time analysis:

For traditional DFS, O(n+m). For back track, the largest cycle is of length n, the number of cycles for each discovered u is O(n). Therefore, for back tracking and calculating , the time it cost is O(n), for each vertex discovered u, the running time should be , therefore, the total running time is