Challenge 1 *story_arc()*

Adapt your happiness score algorithm from HW2, and combine it with your plotting skills from HW3, to write a function that takes a "story" string as input, and plots a happiness arc for it.

.....

Challenge 2 *pagerank()*

Write a function that uses linear algebra from numpy that takes a (possibly weighted, directed network) as input, and outputs two vectors, one that gives the score for each node, and one vector that ranks the nodes from highest pagerank score to lowest.

.....

Challenge 3 *degree()*

Write a function that takes a network as input, and outputs two vectors, as before, but this time using a degree/strength score (total number/weight of outgoing edges).

.....

Challenge 4 *PageRank vs Degree*

Find a real-world network with at least 100 nodes, and compute both the degree and the pagerank centrality scores. Then, plot them against each other in a scatterplot to examine their relation.

.....

Challenge 5 *number_paths()*

Write a function that takes as input a binary network G, two specified nodes v, w, and a natural number k, and outputs the number of paths of length k from node v to w.

.....