```
% IMPORTING DATA

load("wordVecV.mat")
```

```
% PART A.1: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(V, 'euclidean');

fprintf("Part A.1: Sim1, sim2 for part A (euclidian dist): %d, %d", sim1, sim2);
```

```
Part A.1: Sim1, sim2 for part A (euclidian dist): 8, 7
```

```
% PART A.2: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(V, 'cosine');

fprintf("Part A.2: The closest vectors (l2 angle) are: %d and %d", sim1, sim2);
```

```
Part A.2: The closest vectors (l2 angle) are: 10 and 9
```

```
% PART B.0: Creating a normalized version of V

sums = sum(V, 1);
V_normal = V./sums;
```

```
% PART B.1: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(V_normal, 'euclidean');

fprintf("Part B.1: Closest normalized word vecs (euclidian dist): %d, %d", sim1, sim2);
```

```
Part B.1: Closest normalized word vecs (euclidian dist): 10, 9
```

```
% PART B.2: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(V_normal, 'cosine');

fprintf("Part B.2: Closest normalized word vecs (l2 angle) are: %d and %d", sim1, sim2)
```

```
Part B.2: Closest normalized word vecs (l2 angle) are: 10 and 9
```

```
% Part C.0: Making an f_doc vector = [f_doc(1), ..., f_doc(|W|)]
```

```
f_doc = zeros(length(V), 1);

for i = 1:length(V)
    f_doc(i) = sum(V(i,:)~=0);
end

W = zeros(size(V));

doc_lengths = sum(V);

for t = 1:length(V)
    for d = 1:size(V, 2)
        W(t,d) = (V(t,d)/doc_lengths(d)) * sqrt(log((10)/(f_doc(t))));
    end
end

% PART C.1: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(W, 'euclidean');

fprintf("Part C.1: Closest normalized word vecs (euclidian dist): %d, %d", sim1, sim2);
```

```
Part C.1: Closest normalized word vecs (euclidian dist): 10, 9
```

```
% PART B.2: Which pair of vectors has the smallest
%           Euclidian distance?

[sim1, sim2] = get_dists(W, 'cosine');

fprintf("Part C.2: Closest normalized word vecs (l2 angle) are: %d and %d", sim1, sim2)
```

```
Part C.2: Closest normalized word vecs (l2 angle) are: 10 and 8
```

```
function [sim1, sim2] = get_dists(mat, metric)
    euclidian_distances = pdist(transpose(mat), metric);
    dist_mat = squareform(euclidian_distances);

    min_dist = min(euclidian_distances);
    min_dist = min_dist(1);

    [i, j] = find(dist_mat==min_dist);

    sim1 = i(1);
    sim2 = j(1);

%     fprintf("Part A.1: The closest vectors (l2) are: %d and %d", sim1, sim2);
end
```