# CSC 485 A2

## Question 1

### Stemming

The stemmed version, with stop words removed, of the first document is (the porter stemmer was used):

phrase fight cat dog reflect natur tendenc relationship two speci antagonist two speci friend

The stemmed version, with stop words removed, of the second document is:

Dog cat bad relationship

The stemmed version, with stop words removed, of the third document is:

Cat furri

The stemmed version, with stop words removed, of the fourth document is:

Dog man s best friend

### Inverted index posting

"cat": [234569, 234578, 234839] "dog": [234569, 234578, 234879]

### Compressed posting list for "dog"

To compress the posting list for "dog", we can use delta encoding, which stores the difference between consecutive document IDs instead of the absolute values.

The compressed posting list for "dog" is: [234569, 9, 301]

### Cosine similarity

Query: cat dog

#### Doc 1

Cosine similarity $= \frac{\text{Dot product of vectors}}{\text{Magnitude of vector1}\times\text{Magnitude of vector2}}$

Cosine similarity (doc1, Query) $= \frac{1\times1+1\times1}{\sqrt{1+1}\times\sqrt{1+1}} = \frac{2}{\sqrt{2}\times\sqrt{2}} = 1$

#### Doc 2

Cosine similarity (doc2, Query) $= \frac{1\times1+1\times1}{\sqrt{1+1}\times\sqrt{1+1}} = \frac{2}{\sqrt{2}\times\sqrt{2}} = 1$

#### Doc 3

Cosine similarity (doc3, Query) $= \frac{1\times1+0\times1}{\sqrt{1+0}\times\sqrt{1+1}} = \frac{1}{\sqrt{1}\times\sqrt{2}} = \frac{1}{\sqrt{2}} \approx 0.707$

#### Doc 4

Cosine similarity (doc4, Query) $= \frac{0\times1+1\times1}{\sqrt{0+1}\times\sqrt{1+1}} = \frac{1}{\sqrt{1}\times\sqrt{2}} = \frac{1}{\sqrt{2}} \approx 0.707$

# Question 2

## A: Finding the largest integer

### Map function

1. Read each integer in the input split
2. Emit a key value pair with a fixed key, let that key be 'num'

### Reduce function

1. Receive all key value pairs with the same key (in this case, all of them have the same key)
2. Iterate through all the values to find the max (e.g by setting a "max" variable that gets updated every time a larger number is seen)
3. Emit a key-value pair with the key "max" and the value we found to be the largest

## B: Finding the average

### Map function

1. Read each integer in the input split.
2. Emit a key-value pair with a fixed key (e.g., "num") and the integer as the value.

### Reduce function

1. Receive key-value pairs with the same key (all pairs will have the same key).
2. Compute the sum of all the integers and the total number of integers.
3. Emit a key-value pair with the key "average" and the computed average (sum / total count) as the value.

## C: Emit a list without duplicates

### Map function

1. Read each integer in the input split.
2. Emit a key-value pair with the integer as the key and a fixed value (e.g., "1").

### Reduce function

1. Receive key-value pairs with the same key (all pairs with the same integer).
2. Emit a key-value pair with the unique integer as the key and the fixed value "1" (or the integer itself) as the value.

## D: Find the number of unique numbers

### Map function (same as part C)

1. Read each integer in the input split.
2. Emit a key-value pair with the integer as the key and a fixed value (e.g., "1").

### Reduce function

1. Receive key-value pairs with the same key (all pairs with the same integer).
2. Maintain a counter for the distinct integers.
3. For each group of key-value pairs with the same key, increment the counter by 1.
4. Emit a key-value pair with the key "distinct_count" and the final counter value as the value.

# Question 3

## A: Bag Union

### Map function

1. Read each tuple from input splits of bags R and S.
2. Emit a key-value pair with the tuple as the key and the count as the value.

### Reduce function

1. Receive key-value pairs with the same key (all pairs with the same tuple).
2. Compute the sum of the values (counts) for the same tuple.
3. Emit a key-value pair with the tuple as the key and the computed sum as the value.

## B: Bag Intersection

### Map function

1. Read each tuple from input splits of bags R and S.
2. Emit a key-value pair with the tuple as the key, the count as the value, and an additional flag to indicate the source bag (e.g., "R" or "S").

### Reduce function

1. Receive key-value pairs with the same key (all pairs with the same tuple).
2. Identify the values for the tuple in bags R and S.
3. Compute the minimum count of the tuple across both bags.
4. Emit a key-value pair with the tuple as the key and the minimum count as the value.

### C: Bag difference

#### Map function

1. Read each tuple from input splits of bags R and S.
2. Emit a key-value pair with the tuple as the key, the count as the value, and an additional flag to indicate the source bag (e.g., “R” or “S”).

#### Reduce function

1. Receive key-value pairs with the same key (all pairs with the same tuple).
2. Identify the values for the tuple in bags R and S.
3. Compute the difference in counts (R_count - S_count) for the tuple. If the result is positive, include it in the output.
4. Emit a key-value pair with the tuple as the key and the computed difference as the value, if the difference is positive.

## Question 4

### Code

Can be found in q4.py

### Output

Can be found in q4output.txt