1. For the queries below, can we still run through the intersection in time O(x + y), where x and y are the lengths of the postings lists for Sapienza and Rome? If so, give an algorithm that computes the resulting list in time O(x+y).

a. Sapienza AND NOT Rome

b. Sapienza OR NOT Rome

2. We have a two-word query. For one term the postings list consists of the following 16 entries:

[4,6,10,12,14,16,18,20,22,32,47,81,120,122,157,180]

and for the other it is the postings list:

[11, 12, 47].

Work out how many comparisons would be done to intersect the two postings lists with the following two strategies. Briefly justify your answers:

a. Using standard postings lists

b. Using postings lists stored with skip pointers, with a skip length of $\sqrt{P}$, as suggested in the class

3. Show that, for normalized vectors, Euclidean distance gives the same proximity ordering as the cosine measure.

4. Show how we can compress the list

[3, 17, 40, 43, 54, 60] using

a. Variable byte encoding

b. γ encoding

5. The following list of R's and N's represents relevant (R) and nonrelevant (N) returned documents in a ranked list of 20 documents retrieved in response to a query from a collection of 10,000 documents. The top of the ranked list (the document the system thinks is most likely to be relevant) is on the left of the list. This list shows 6 relevant documents. Assume that there are 8 relevant documents in total in the collection.

R R N N N   N N N R N   R N N N R   N N N N R

a. What is the precision of the system on the top 20?

b. What is the recall on the top 20?

c. What is the F_1 measure on the top 20?

d. What is the uninterpolated precision of the system at 25% recall?

e. What is the interpolated precision at 33% recall?