CS 345: Homework 5 (Due: Dec 2, 2013 before 10:01am)

Rules. This is to be completed no later than the start of class on the day it is due. Earlier submission is also possible by email (read Handout 1).

```
Document 1 (docID 1):
A sentence is a group of words. Sentences have a subject and a verb
and maybe an object.

Document 2 (docID 2):
In a sentence find the verb and then find the subject.

Document 3 (docID 3):
To find the subject ask who or what followed by the verb.

Document 4 (docID 4):
To find the object ask  'subject verb who or what'?
```

Problem 1. (6 points) **(This is a followup of Problem 1 of HW1)**
By providing screenshots, or other hard evidence explain how often your weg-page gets crawled by Google. Be reminded that in Problem 1 of Homework1 you setup the framework to collect this information. You might still have time to do so between the time of the posting of this homework (early november) and the due time (early december).

Problem 2. (12 points)
(a) Provide v-codes for $31,257$ in hexadecimal.
(b) You are given an inverted list for a term t that looks like

```
(1,1)(1,5)(1,11)(1,20)(2,10)(2,18)(2,30)(3,11)(3,25)(4,90)(4,91)
```

Use the discussion of Subject 7 to v-byte encode this inverted list by providing the following information. (Note that v-byte encoding is also discussed on 150 of the textbook. However, the example of the textbook incorrectly applies gap encoding to a count field shown here in bold-face, contradicting the algorithm stated there or in the notes: *1,2,1,6,1,***2***, 6, 11, etc.*)
(b1) Give the flattened (no parenthesis) form of the to be v-byte encoded list just before the v-byte encoding is applied when the list is flattened but still in decimal notation, and
(b2) after the v-byte encoding has been applied and the list given in hexadecimal notation.

Problem 3. (18 points)
Using the following stopword list:
`an a and in is of or the then to who what`,
for the four documents shown above, show the form of the occurrence lists if (a) Doclist is used, (b) Counts is used, (c) Positions is used.
(d) For Positions, in the previous problem, what would an implementation of vocabulary and the inverted list table look like ?

Problem 4. (8 POINTS)
Use the best approach possible to evaluate a query
`tA AND tB AND tC AND tD AND tE`
where the doclists of `tA,tB,tC,tD,tE` are of length $1000, 15, 2000, 10, 10000$ respectively. What is the total number of operations performed? Explain and express the answer in terms of the length of the doclists.

Problem 5. (12 POINTS)

Use Homework 2 and queries Q2 for Google and Bing. Printouts are available on the Web (more readable than the Homework 2 hardcopy). The solutions for Homework 2 contain relevant and non-relevant documents.

(a) Provide 3-point effectiveness rates and 3-point averages of those rates separately for each one of the queries/engines. (Note: Use the methods of Subject 8 as needed if incomplete information is available.)

(b) The fill in the tabulation table below. Think of it as an extension to the one used in Homework 2. Also fill the last-column with who wins, who loses and tie points. (Note that the interpolated precision figures asked might not be the ones computed from part (b). "Interpolate" appropriately as it was done in Subject 8 and in class.)

	Values		Points	
	Google	Bing	Google	Bing
# Retrieved docs for Q2	10	10	1	1
# Relevant docs for Q2	9	6	1	0
20% recall interpolated precision for Q2				
50% recall interpolated precision for Q2				
80% recall interpolated precision for Q2				
3-point effectiveness(20,50,80) for Q2				
Number of point wins (sum)	-	-		

Problem 6. (10 POINTS)

Apply the HITS algorithm and PageRank algorithm on the following graph, and determine the ranking values (authority/hub for former, and rank for latter). Iterate as many times as needed for the error to be less than 10^{-3}. Use the synchronous update version.

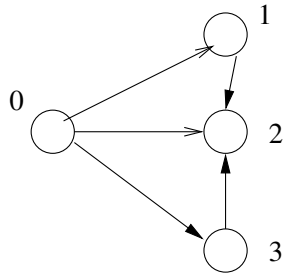

Figure 1: Graph for HITS and PageRank computation.

Problem 7. (24 POINTS)
This is a continuation of Problem 5 of HW4. So read the problem and its solutions if you did not solve it completely and correctly before you pursue this problem further. (I even use indexing for the questions that continues the index of HW4.P5.)

Q5 (2 points). What percentage of the machines become query machines (use definition before text of Q8)? Justify your answer and make sure it makes sense given the load of a cluster.

Q6 (4 points). Based on the answers of Q1-Q5 (round up to closest integer)

6.1 What is the average document size.

6.2 What is the average index-size contribution per document.

6.3 What is the average DocIndex contribution per document.

6.4 Average links contribution per document.

Q7 (4 points). Based on Q6

7.1 What is the average size per document not only of the raw document but also of doc-index data structures such as thos mentioned in 6.1-6.4.

7.2 What is the average size per document of all index-related data structures as mentioned in 6.1-6.4

7.3 What is the document / index-ratio as can be derived from 7.1 and 7.2 (no rounding)?

7.4 If one maintains 2 copies of the raw documents, and 4 copies of the index, what is the modified answer?

You are given 2000 machines (PCs). Using the data available in the paper answer the following questions in a way that is fully compatible with the paper. Read all the questions first. You may assume that you will use only three types of servers: index, doc, and query servers (that will become GWS, cache, ad, spell check ones).

Q8 (3 points). How would you organize the 2000 machines in racks etc? (Note keep a backup of roughly 4% of resources. Also assume that switches are mostly 100Mbit with 2 additional Gigabit uplinks each. Few Gigabit switches are also available.)

8.1 How many racks are you going to use and how many rack-sides are there?

8.2 How many network switches?

8.3 Other resources? Explain.

Q9 (2 points). What is the collective memory (RAM), disk space (TBytes) of these resources? And also per rack (GBytes)!

Q10 (3 points). Using data from other papers you have already read for this course,

- How many documents can your cluster design support? (Be consistent with Q7 on doc/index size.)
- How many of these machines will be file servers, how many index servers, and how many query machines? (Be consistent with Q7 on doc/index size.)
- How many queries can the structure support per second?

With respect to the Baroso etal paper (Google cluster architecture) answer, by quoting text of the paper, the following questions.

(Q11) (3 points) In 2003 (based on the paper) did Google store compressed or uncompressed Web data?

(Q12) (3 points) In 2003 (based on the paper) did Google store the index itself compressed or not?