NJIT New Jersey's Science & Technology University

A. V. Gerbessiotis | CS 485-101

December 2010 | Fall 2010

Exam 2(150 points) | 90 minutes

Name: .....

ID Number: ..... **Exam Number:** .....

Grade: 1: ... 2: ... 3: ... 4: ... 5: ... Total: .....

Solve **ALL the problems** in the space provided

Read the Problems CAREFULLY!

There are 7 (SEVEN) pages this page included

Read and Sign the statement below at the end of the exam

Unsigned exams will be marked with grade 0 (zero).

STATEMENT

On my honor, I pledge that I have not violated the provision of the NJIT Student Honor Code.

Sign below at the end of the exam

Signature .....

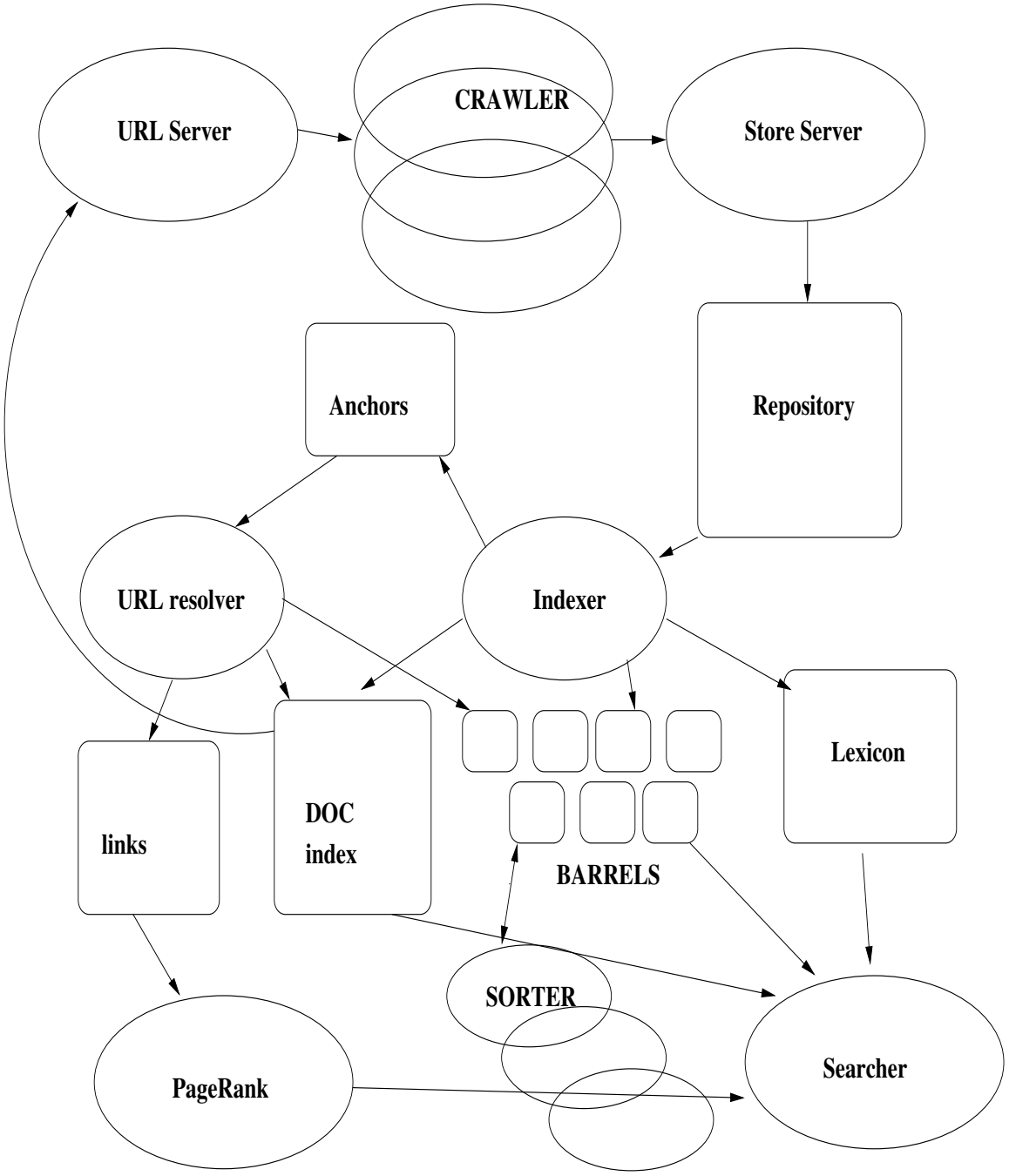

Figure 1: Google System Architecture.

Problem 1. (35 POINTS)
Give short answers to the following questions. (Use the Google System Architecture figure if you need it, for references related to Google.)

(1) Spider, bot, robot all refer to the same thing. Give another name for it.

(2) What is the input for the ranking computations in Google (1997/1998)?

(3) In its inverted index does Google store URLs or `docID`s?

(4) What do we call a frequently occurring word in a text that is not a keyword?

(5) What is the difference between a keyword and index-term? Explain and provide an example.

(6) A user types in the query `NOT A B` in Google. What documents is the user interested in recovering? Explain.

(7) A user types in the query `NOT A B` in Bing. What documents is the user interested in recovering? Explain.

Problem 2. (25 POINTS)
For a specific query involving term t, the corpus (collection of documents) includes the ten documents $d_1, \ldots, d_{10}$. Five of them, $d_1, d_3, d_5, d_7, d_{10}$ contain index term t and are relevant, whereas the remaining five d_2, d_4, d_6, d_8, d_9 are not relevant.
As a response to the query Search Engine NJIT-V1 returned documents

$$\mathbf{d_1}\mathbf{d_3}d_8d_4\mathbf{d_7}d_9$$

in that specific order.
(a) What is the precision of the search query? Explain.
(b) What is the recall of the search query? Explain.

Problem 3. (30 POINTS)
Continuing from Problem 3 the same query involving term t was input to NJIT-V2, a newer version of the engine. NJIT-V2 was buggy so it spit out everything.

$$\mathbf{d_1} d_2 \mathbf{d_3} d_4 \mathbf{d_5} d_6 \mathbf{d_7} d_8 d_9 \mathbf{d_{10}}$$

(a) What is the precision of the search query (after all the outputs are displayed)? (5 points)

(b) What is the recall rate for the query (after all the outputs are displayed)? (5 points)

(c) Develop a table with recall and precision rates starting with the first document d_1 all the way to the last d_{10}. (10 points)

(d) Develop a 6-point table with a recall step at 20% . Then compute the 3-point effectiveness. (10 points)

Note. Use `2/3 = 66% , 3/7 = 42% , 4/7 = 56% , 4/9 = 44% , 5/9 = 55%` . If you know interpolated precision at recall rate 20% (say it is 30%) and at 40% (say it is 50%) and you want to compute it at 30% recall, interpolate, i.e. take the average $(30 + 50)/2 = 40$.

Problem 4. (30 POINTS)
(a) You have a machine with 1000MBytes of free memory. We want to use all this memory to store a hash table representing the lexicon of a search engine. The average length of a word w stored in the lexicon is 8 ASCII characters. The hash table will store a pointer p that points to a location in memory where the NULL terminated string for word w is stored; a pointer p is 4 bytes long. In addition to p a `wordID` needs to be locatable in addition to w. Organize the hash table so that you extract from it maximum performance. How many words n can the lexicon accommodate? Explain. What is the length of a `wordID` (do not underestimate, but overestimate up to the next available byte).
(b) Give the unary and Elias-γ codes for 4.

Problem 5. (30 POINTS)
Assumptions. For the following question you must use the following cost estimates.

- **Cost 1.** The merging of a list of n sorted elements with a list of m sorted elements has cost $n + m$.
- **Cost 2.** Scanning a list of n elements and performing a constant time operation will cost n.
- **Cost 3.** Sorting a list of n elements costs $n \lg n$.
- **Cost 4.** Binary search on a list of n sorted elements looking for element x costs $\lg n$.

We have three (sorted by `docID`) doclists for terms `A,B,C` of length $10, 256, 1024$ respectively. We would like to perform the following query:
`A AND B AND C`
What would the cost of an efficient evaluation be? Use the cost metrics Cost 1,2,3,4 in your analysis.

Intentionally left blank. You can use it for scratch paper by tearing it off carefully; do not return it then.

End of Exam 2