

Name:

ID Number:

Exam Number:

Grade: 1: ... 2: ... 3: ... 4: ... 5: ... 6: ... 7: ... 8: ... Total:

SOLVE **ALL the problems** IN THE SPACE PROVIDED

Read the Problems CAREFULLY!

THERE ARE 8 (EIGHT) 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

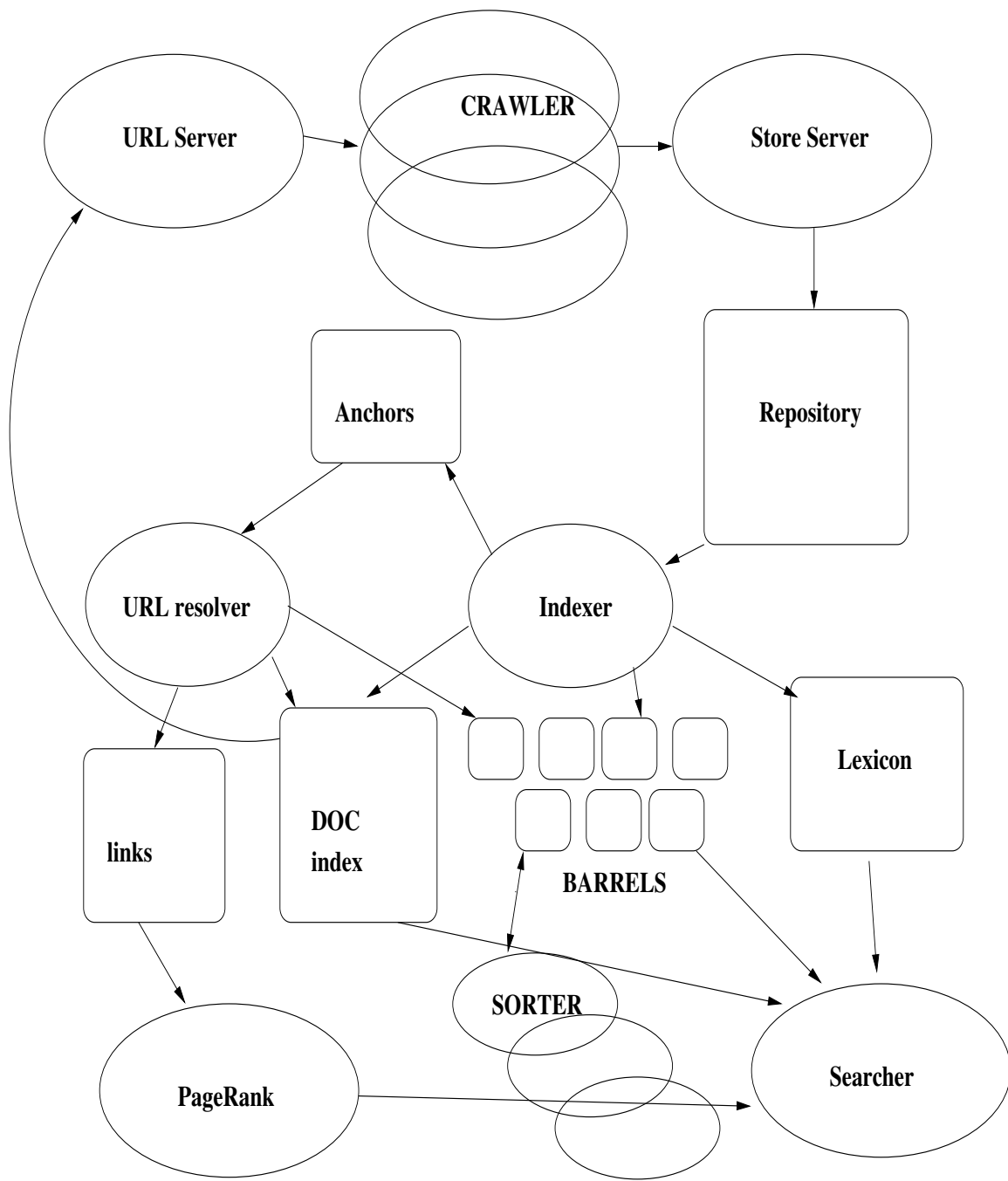


Figure 1: Google System Architecture.

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) Where are web-pages stored in Google?
- (3) Do long barrels store long URLs, short URLs or neither?
- (4) Give a data structure in Google that stores only **docIDs** and neither URLs nor URL checksums.
- (5) What is the input for the ranking computations in Google (1997/1998)?
- (6) In Google 1997/1998 why does the Searcher access the Doc Index?
- (7) We have the **Web**, we have the **Semantic Web**, the **Surface Web**, what other Web do we have?

Problem 2. (35 POINTS)

TRUE or **FALSE**? Only a fully written **TRUE** or **FALSE** qualifies as a valid answer. Alternatives, such as **T** or **F** would be considered **WRONG**!

- (1) Google's StoreServer stores in it compressed crawled pages.
- (2) The Indexer stores the forward index in the DOC index.
- (3) Links contains full-path URLs of web-pages.
- (4) PageRank computations use link structure described in terms of **docIDs**.
- (5) PageRank computations use link structure described in terms of **URLs**.
- (6) The acronym **URL** means Uniform Resource Language.
- (7) **URL** (see part (6)) is a subset of **UML**.

Problem 3. (40 POINTS)

For a specific query involving term t , the corpus contains ten documents $d_1, \dots, d_{10}$. Five of them, $d_1, d_2, d_5, d_8, d_{10}$ contain index term t and are relevant, whereas the remaining five are not relevant.

d₁d₂d₃d₄d₅d₆d₇d₈d₉d₁₀

- (a) What is the precision of the search query (after all the outputs are displayed)? (5 points)
- (b) Develop a table with recall and precision rates starting with the first document d_1 all the way to the last d_{10} . (15 points)
- (c) Develop a 6-point table with a recall step at 20% . Then compute interpolated precision. Then compute the 3-point 20-50-80 effectiveness. (20 points)

Note. Use integer results only and percentages. So $2/3 = 66\%$, $3/7 = 42\%$, $4/7 = 56\%$, $4/9 = 44\%$, $5/9 = 55\%$.

Problem 4. (30 POINTS)

- (a) Give the unary, binary, and Elias- γ codes for 5.
- (b) We have the string of 20 characters A, C, G, T shown below.

TTTT AAGG TTCC AAAA AAGG

(Ignore spaces, they are provided for clarity only.) If you compress this text using Huffman coding, what are the prefix codes for each one of the four characters? What is the total number of bits used to encode just the text (consisting of A, C, G, T and ignoring the spaces or other auxiliary information)?

Problem 5. (30 POINTS)

We typed the following queries in Google and Bing and recorded the following number of hits in millions for the two search engines. Answer the following questions with respect to the Table below.

No	Query Term(s)	GOOGLE (number of hits in '000,000)	BING (number of hits in '000,000)
1	$\langle a \text{ b or } c \rangle$	2530	2680
2	$\langle a \text{ b OR } c \rangle$	22100	5150
3	$\langle A \text{ B OR } C \rangle$	22100	5150
4	$\langle a \text{ (b OR } c) \rangle$	25270	6870
5	$\langle (a \text{ b }) \text{ OR } c \rangle$	22100	5150
6	$\langle a \text{ or } b \rangle$	3230	3090
7	$\langle a \text{ OR } b \rangle$	25270	11500
8	$\langle b \text{ or } a \rangle$	3270	73
9	$\langle b \text{ OR } a \rangle$	25270	11500
10	$\langle a \text{ not } b \rangle$	2910	1450
11	$\langle a \text{ NOT } b \rangle$	2910	0
12	$\langle a \text{ NOT } a \rangle$	19960	0
13	$\langle a \text{ - } a \rangle$	0	0

(a) With reference to queries 1-5 which queries are equivalent? Explain. Which queries does Google and/or Bing consider to be equivalent? Explain.

(b) Is there a difference between OR and or for the two engines? Explain.

(c) Is there a difference between NOT and not for the two engines? Explain.

(d) What does query 12 ask? Explain.

(e) What does query 13 ask? Explain.

Problem 6. (25 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.** The merging of a list of n sorted elements with a list of m sorted elements and elimination of duplicates takes $n + m$.
- **Cost 3.** The merging of a list of n sorted elements with a list of m sorted elements and elimination of nonduplicates takes $n + m$.
- **Cost 4.** Binary search on a list of n sorted elements looking for element x costs $\lceil \lg(n + 1) \rceil$.

We have three (sorted by `docID`) doclists for terms `A,B,C` of length 100, 1000, 100 respectively. We would like to perform the following query: What would the cost of an efficient evaluation be? Use the cost metrics provided.

`A OR B OR C`

Problem 7. (30 POINTS)

You are given an inverted list for a term t that looks like

`(1,1)(1,2)(1,8)(1,12)(2,8)(2,12)(2,16)(2,20)(3,15)(3,20)(4,1)`

Use gap encoding to encode this list. For a document x , its list would be converted from the doclist form $(x, h_1), \dots, (x, h_y)$ into a one that looks like $(\bar{x}, \bar{y}, [a_1, \dots, a_y])$, where $\bar{x}$ is the gap-encoded form of docID x , $\bar{y}$ is that for y , and $a_1, \dots, a_y$ are the forms of the hits $h_1, \dots, h_y$. (See also page 150 for a discussion but beware of the typos there.) That is we need the non-flattened form (with parenthesis/brackets) before parentheses are removed, and before v -byte codes are determined. All numbers in decimal.

Problem 8. (25 POINTS)

A corpus contains 10^8 ASCII-text documents. The average size in bytes per document is $100KB$, and the average length of a word inclusive of a single whitespace separator is 10. If Heaps' Law in the form $n = 40N^{0.5}$ is applicable, where n is the number of distinct words stored in the dictionary and N is the total number of words in the collection, how large would the dictionary be for such a corpus. Express the answer in MB. To build a dictionary assume the following

- Distinct words are stored NULL separated but in UNICODE in an array of contiguous words.
- A hash table structure is used to contain a pair that includes a 4-byte pointer to the array of contiguous words, a `wordID` identifier (you specify the minimum number of bytes), and organized so that it can be efficiently accessed.

What would the size of the dictionary (contiguous array plus hash table) be in MB? Explain.

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

End of Exam 2