

EECS 498 System Design of a Search Engine

Winter 2026

Lab 7: Midterm Review

GSI: Christopher Davis
IA: Rickey Dong

Nicole Hamilton
[https://web.eecs.umich.edu/~nham/
nham@umich.edu](https://web.eecs.umich.edu/~nham/nham@umich.edu)

agenda

1. Announcements
2. Midterm Info
3. Midterm W25

agenda

1. Announcements
2. Midterm Info
3. Midterm W25

announcements

Reminder:

- **LinuxGetSSL** and **LinuxTinyServer** are due this Sunday (2/22)
- No lab next Friday (2/27) – have a good Spring Break!

Recommended Resources for Networking:

- Lectures 10-12 on Canvas
- [Beej's Guide to Network Programming](#)

agenda

1. Announcements
2. Midterm Info
3. Midterm W25

midterm details

- Midterm will be in-class **next Thursday** (2/26)
- Paper exam
- No cheat sheet
- Covers until the Index & Constraint Solver (2/24 lecture)
- Review both lectures *and* assignments
- No content on developer tools stuff (CMake, GTest), maybe something from last week's lab demo
- Short response questions
- There may be a coding question

midterm review

Midterm review slides on Canvas:

- [Exams/Midterm/Midterm Review.pdf](#)

Office Hours

- Mon: 3:30-4:30pm - 1637 BBB, Table #1 (Rickey)
- Tues: 12-3:00pm - 1637 BBB, Table #2 (Chris)
- Tues: 3-4:00pm - 4820 BBB (Prof. Hamilton)
- Wed: 2:30-4:00pm - 1270A Leinweber (Rickey)
- By appointment/zoom: email chcdavis@umich.edu

agenda

1. Announcements
2. Midterm Info
3. **Midterm W25**

midterm W25

List the major parts of a search engine.

midterm W25

List the major parts of a search engine.

1. HTML parser.
2. Crawler.
3. Inverted word index.
4. Constraint solver.
5. Query compiler.
6. Ranker.
7. Front end.

midterm W25

One of the reasons UTF-8 became popular is because it's backward compatible with what?

midterm W25

One of the reasons UTF-8 became popular is because it's backward compatible with what?

ASCII.

midterm W25

When a Linux process forks, the parent and child initially share their memory pages. But if one of them makes a change, what happens? What's that called?

midterm W25

When a Linux process forks, the parent and child initially share their memory pages. But if one of them makes a change, what happens? What's that called?

The shared pages are initially marked read-only. When a process tries to write to a RO page, that causes a hardware interrupt into the operating system, which makes a copy of the page, marks them both RW and gives one to the parent, the other to the child. This strategy is called copy-on-write.

midterm W25

When a process forks, how does it know if it's the parent or the child?

midterm W25

When a process forks, how does it know if it's the parent or the child?

fork() returns 0 for the child process, and a positive int for the parent process (the PID of the new child process).

midterm W25

You have two threads sharing two resources, A and B. To use a resource, you must first lock it, then release the lock when you're done. Thread 1 needs A and sometimes B as well, so it locks A and then, when it needs it, it locks B also. Thread 2 needs B and sometimes A, so it locks B and then, when it needs it, it locks A. Will this work? If there's a problem, what must be done to fix it?

midterm W25

You have two threads sharing two resources, A and B. To use a resource, you must first lock it, then release the lock when you're done. Thread 1 needs A and sometimes B as well, so it locks A and then, when it needs it, it locks B also. Thread 2 needs B and sometimes A, so it locks B and then, when it needs it, it locks A. Will this work? If there's a problem, what must be done to fix it?

No, this will not work. It will deadlock. To fix the problem, if multiple resources are needed, they must always be requested in the same order. Here, it must be either A, then B, or B then A. If the order is A then B, then for thread 2, already holding B, to acquire A, it must first release B, then lock A, then B again. Not technically required but considered good practice is releasing them in reverse order to that in which they were taken.

midterm W25

List the steps to reading a webpage using HTTP.

midterm W25

List the steps to reading a webpage using HTTP.

1. Determine the IP address and port number for the server.
2. Create a socket.
3. Connect it to the server.
4. Create an appropriate GET message for the page you want.
5. Write the GET message to the host over the socket.
6. Read the response and keep reading until there is no more.
7. Close the socket.

midterm W25

To serve up pages over HTTP, what does a simple web server do?

midterm W25

To serve up pages over HTTP, what does a simple web server do?

The server creates a socket, binds it to its IP address and port number, and begins listening. It then loops, accepting all connect requests, passing each request off to a separate thread to handle the expected GET request.

midterm W25

How do you set up a memory-mapped file?

midterm W25

How do you set up a memory-mapped file?

Open the file, then pass the file handle or descriptor to the operating system function `mmap( )` asking that it be mapped into your address space. It returns a pointer to the beginning of where it's been mapped.

midterm W25

What is the advantage of a memory-mapped file?

midterm W25

What is the advantage of a memory-mapped file?

There's no need to do individual file reads or writes to/from buffers. Instead, the whole file is mapped into the process virtual memory address space. To read/write any location in the file, all you have to do is dereference a pointer into that address range where the file has been mapped. When you read or write addresses in that range, the paging system is used to fetch pages from disk as needed. The file system caching mechanism will try to keep the most used blocks in memory without us having to do anything to make that happen.

midterm W25

If you would like to be able to search for “amazon” only in the title in an inverted word index, how would you do that? If you do it that way and you now want to find pages that have “amazon” anywhere, what would you have to do?

midterm W25

If you would like to be able to search for “amazon” only in the title in an inverted word index, how would you do that? If you do it that way and you now want to find pages that have “amazon” anywhere, what would you have to do?

You could decorate the term with characters you expect to strip out from anything a user would type during query parsing to distinguish the posting list for amazon in each of these four cases. You get to pick any characters you like as long as they’re characters you expect to strip. For example, you might index “@amazon” for occurrences in anchor text, “=amazon” for URL, “+amazon” for title and unadorned “amazon” for body. If you do this, then to search for “amazon” anywhere, you would have to OR (merge) the individual posting lists.

midterm W25

Why would you need a seek index or synchronization table for a posting list?

midterm W25

Why would you need a seek index or synchronization table for a posting list?

The posting list is typically compressed both by storing deltas from the previous post rather than absolute locations and by a variable-length encoding scheme, e.g., UTF-8. To know the absolute location of post, you have to add up all the deltas starting from some known location, either the beginning of the index or an entry in a seek index. The seek index could be an array, perhaps indexed by high bits of a seek address, or a list of known locations. Each entry is a pointer to a post, if there is one, and the location where it occurs.

midterm W25

What does a bag of words refer to and why is it not a very good ranking technique?

midterm W25

What does a bag of words refer to and why is it not a very good ranking technique?

It's a ranking technique that considers the query and the document as two unordered lists of words and attempts a measure of similarity based on a comparison of word frequencies. The more hits in the document of the rarer words in the query, the better. It's not a very good technique because it does not consider where the hits were found or their proximity to each other, e.g., as a phrase, or near the top of the page, and tends to be easily spammed, e.g., by keyword stuffing.

midterm W25

What happens if you try to seek way past the end of a file and begin writing?

midterm W25

What happens if you try to seek way past the end of a file and begin writing?

The operating system will let you do it, possibly creating a “hole” in file where there’s no data.

midterm W25

What is the difference between precision and recall? Which is more important for a search engine?

midterm W25

What is the difference between precision and recall? Which is more important for a search engine?

Precision is the fraction of the results that are relevant. Recall is the fraction of all relevant results that are returned. Precision is much more important for a search engine because no one ever wants to see all the matching pages, only the best.

midterm W25

What is the difference between static rank and dynamic rank?

midterm W25

What is the difference between static rank and dynamic rank?

Static rank is a measure of how good the page is without regard to the query. Dynamic rank is a measure of how good a match the page is for the specific query.

midterm W25

You can create a thread using `pthread_create()`. But what does `pthread_join()` do?

midterm W25

You can create a thread using `pthread_create()`. But what does `pthread_join()` do?

`pthread_join()` waits until the specified thread ends and retrieves the exit status.

midterm W25

What is a sparse file?

midterm W25

What is a sparse file?

A file with a “hole” in it where there’s no data. Reading from the hole returns zeros.

midterm W25

Traversing a directory tree, which directory entries would you typically skip?

midterm W25

Traversing a directory tree, which directory entries would you typically skip?

“.” and “..”, referring to the current directory and the parent directory.

midterm W25

If your index is broken into chunks with disjoint (non-overlapping) slices of the web you've crawled, what factors would you consider in deciding whether to search them sequentially or concurrently with separate threads when a query arrives?

midterm W25

If your index is broken into chunks with disjoint (non-overlapping) slices of the web you've crawled, what factors would you consider in deciding whether to search them sequentially or concurrently with separate threads when a query arrives?

If multiple chunks are on the same physical rotating disk drive, you might prefer sequentially to avoid chattering the heads back and forth between the two index chunks. If they're on separate mechanical drives or on solid state drives (which seek times are minimal), you might prefer multiple threads. If the chunks are tiered with best pages sorted into a small index, you might start there before deciding whether to bother searching the other chunks.

midterm W25

What are some ways you might try to shorten the time it takes process a query? Would any of these methods likely harm the quality of the results?

midterm W25

What are some ways you might try to shorten the time it takes process a query? Would any of these methods likely harm the quality of the results?

The most significant determinant of query time is the number of matching pages that must be ranked. There are lots of ways to limit this.

- a. Limit the total size of the index on any given machine by adding more machines to the cluster
- b. Stop searching for new matching pages once a maximum number have already been found and ranked or after a maximum elapsed time.
- c. Create a small index of presumed high-quality pages and a larger index of everything else. If you get enough good results in the small index, quit there; don't search the large index. If you do have to search the large index, presumably it's because there aren't many matches, in which case, the big index search could be fairly quick.
- d. Incrementally broaden the search. Search first for matches in titles (or URLs or anchor text), before deciding whether to broaden the search for pages that only match in body text.

These strategies are unlikely to significantly harm result quality. If a query is matching lots of pages, it suggests that many are probably of roughly equal quality. Not returning the one very best won't be noticed.

midterm W25

What types of search engine queries are likely to be slowest?
Would lots of words in the query make the search faster or slower? Why?

midterm W25

What types of search engine queries are likely to be slowest?
Would lots of words in the query make the search faster or slower? Why?

The slowest will be the one-word queries for common words, e.g., a query for "the". They'll match an enormous number of documents, forcing each to be ranked. Adding more words to the query, especially if some are rare, will make the search faster because it won't match as many documents, meaning there won't be many that need to be ranked.

midterm W25

You have two asynchronous activities that need to run concurrently. How would you decide whether to make them separate threads or separate processes?

midterm W25

You have two asynchronous activities that need to run concurrently. How would you decide whether to make them separate threads or separate processes?

It would depend how tightly coupled and similar they are and whether there might be some security issues. If it's lots of the same activity over-and-over, e.g., reading web pages, that would work well as a thread. But if the activities are not tightly connected, e.g., building the index vs. serving queries, or if they're handling sensitive information, e.g., handling credit card numbers, the firewalling of doing these as separate processes would make that more attractive.

midterm W25

Would it be okay if errors in the HTML cause your search engine to misparse a page?

midterm W25

Would it be okay if errors in the HTML cause your search engine to misparse a page?

Yes, if the HTML was broken, the page was probably lower quality anyway. If you miss an occasional word or link, especially because of broken HTML, it's unlikely that will affect search results in noticeable way.

midterm W25

If you look up the IP address of a host, then connect to it, why do you still need to specify the host again in a GET request? You already know it's the right machine based on the IP address.

midterm W25

If you look up the IP address of a host, then connect to it, why do you still need to specify the host again in a GET request? You already know it's the right machine based on the IP address.

Any given machine might be hosting several websites at the same IP address and port number.