Quora uses cookies to improve your experience. Read more

Related Questions

Threading (computer science)   Programming Interviews   +4

What are the programming interview questions related to multi-threading/synchronization in c?

✎ Answer   🔊 Follow · 20   ➜ Request   ◯ ⤓ f 🐦 ➹ •••

5 Answers

Terry Lambert, Apple Core OS Kernel Team; technical lead on several projects over 8 years
Answered Mar 3, 2017

A2A

Sorry, but this is *really* an overly-broad question.

I will mostly not ask threading/synchronization questions, unless something specific in the interview leads us down that particular rabbit hole. And then the questions will depend on the path we took to get there.

I could easily come up with 500 questions about various things about threading.

If you wanted to take computer language and/or platform specifically into account, I could, with difficulty, come up with perhaps 5,000 — but it'd take me a while to do it, and I'd have to drill down on certain topics into some obscure areas.

Here's some threading questions, off the top of my head, just stream of consciousness, with no attempt to organize them into categories.

- When and why do you use threads?

- What are the advantages and disadvantages to a thread over a process?

- What happens when a multithreaded program is preempted by the scheduler?

- Are there kernel and user space pieces to all threads implementations, or can you do it all in one place or the other?

- Can you safely use a recursive mutex in the kernel in Windows as part of a blocking operation for a multithreaded program in user space?

- Is there any difference between a synchronization primitive that has to operate between processes vs. between threads in a single process?

- How do threads interact with file locking?

- When would you use each of these threading models, and why?

  - Free-threading

  - Rental-model threading

  - Apartment model threading

More Related Questions

Question Stats

19 Public Followers

14,114 Views

Last Asked Nov 15, 2017

Edits

    ◦ Thread pooling

- What's a pool mutex, and when and why would you use it instead of some other form of synchronization?

- What's the difference between a mutex and a semaphore?

- What's the difference between a mutex and a lock?

- What's better: recursive locks, or non-recursive locks, and why?

- What's a serialization barrier?

- Is there a difference between serializing data vs. serializing CPU instructions?

- What's an atomic operation?

- What's an idempotent operation?

- How the heck does "errno" work in a multithreaded program?

- What's the difference, if any, between idempotence, and atomicity?

- Explain Djikstra's "Banker's Algorithm"

- Explain the "Dining Philosopher's" problem

- What's the difference between voluntary and involuntary preemption?

- How does a call conversion scheduler work?

- I've always wondered… what precisely is the difference between a thread and a process?

- What's a thread specific variable?

- What's the difference between a thread-safe and a thread-unsafe function?

- What's a critical section?

- Assume you have two threads that can come into the same code, and might try to operate on the same data at the same time; do you lock the data to keep them from doing it, or do you lock the critical section of code to keep them from doing it?

- Explain reference counting, and where you'd use it.

- Explain when you can use an atomic operation on a reference count, and when you have to use a lock on it instead.

- What's a reference alias?

- How do you design an API to prevent a reference alias, if you want to prevent them?

- Explain scoping of pointers to allocated and reference counted objects, and tell me why it matters, or if it doesn't, why not?

- What's a thread exit mean?

- What's a thread join?

- When you create a thread, why would you want to set thread attributes other than the default, on creation, rather than right after, to make the
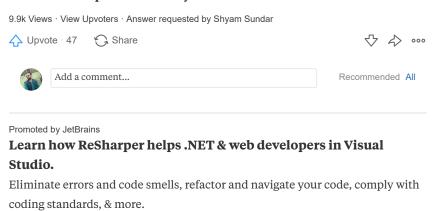
code simpler?

- What happens to your threads when you vfork?

- What happens to your threads when you fork?

- What happens to your threads when you exec?

- What's a "work to do" model?

- What's a "thread per client" model?

- How would you use threads to better implement a web browser?

- What are some security implications to using threads vs. using processes?

- Can one thread cause another thread to crash?

- If one thread can cause another thread to crash, does just the thread crash, or does the whole program crash?

- How many threads should you use in a program that runs on a 4 CPU system, and why?

- What's a dispatch thread?

- When do you use dispatch threads?

- Are dispatch threads used very often anyway?

- Have you ever heard of "sfork"?

- What was the first program in which you ever used threads?

There's 50 questions (53, if you count each of the models as a separate question).

I wrote them down about as fast as I can type. Which is pretty fast. If I kept on much longer, my wrists would start to get sore, if I continued to type that fast.

And yes, before you ask: I can give answers to all of the questions above.

Like I said: the question is overly broad.

9.9k Views · View Upvoters · Answer requested by Shyam Sundar

⬆ Upvote · 47   ↻ Share   ⬇ ⤴ ⚬⚬⚬

[avatar] [Add a comment...   ]   Recommended   All

[avatar] Vaclav Krpec, SW Developer (Since 2003)
          Answered Jan 24

1st, you must know the synchronisation primitives and their properties: atomic operations, mutex, condition variable, RW lock, semaphore, barrier and the operations these allow (e.g. try-locking). You need to understand the difference between a thread and a process, obviously, and what's the difference between parallelism and concurrency. You must know what deadlock and livelock are.

You should know the common stereotypes (general types of parallelism problems); e.g. producer-consumer problem (that one comes almost always at an interview), leader selection, latency-free timing etc.

You should also think about how would you emulate certain sync. primitives using other ones; e.g. RW lock using mutex and perhaps condition variable and the implications of such an implementation (like the possibility of writer starvation with over-simple RW lock implementation).

Note that unlike C++, MT in C is typically platform-dependent. You'll need to look through the primitives the OS in hand provides; namely MT models differ somewhat e.g. between POSIX systems and Windows (although the principles should be very similar).

For scalable networking, you should also take a look at async. socket polling (it's a bit off-topic regarding MT, but closely related).

Last but not least, MT is not a simple thing and requires some experience. Design of a multi-threaded application is something you need to do several times to get used to it and do it justice.

1.7k Views · View Upvoters

⬆ Upvote · 1   ↻ Share   ⬇ ⤴ ⋯

| | Add a comment... | | Recommended All |

Al Klein, 46 years of earning a living developing systems.
Answered Jan 15, 2017

It doesn't work like that. The interviewer will ask questions related to what *he's* interested in (as it relates to the job being interviewed for). If you don't know anything about multithreading, there's no way you can study a fixed set of questions in order to land the job. This isn't a school test, with a fixed number of questions. You might get some question that's seemingly unrelated to computers thrown at you - and that could be the one question that determines whether you get hired or not.

(I used to give a "test" in an assembly language I made up - to compare an input with a list of values. But the CPU had no "compare" instruction. Real programmers had no problem - "compare" is "subtract". People who memorized things couldn't find a compare instruction, so they couldn't write the program. I

wanted someone who *understood programming*, not someone who had memorized answers.)

3.6k Views · View Upvoters

⌃ Upvote · 10　　🔄 Share　　　　　　　　　　　　⌄　⤳　⚬⚬⚬

　　　　　Add a comment...　　　　　　　　　　Recommended　All

Dan Allen, I've been programming for 34 years now. Still not finished.
Answered Mar 3, 2017

In C it will depend a lot on whether you're going to be asked about a specific threading library or whether the employer is willing to cross-train.

There is no standard threading library and given it is a bit of specialism many employers will be interested in people who have an expert knowledge of the principles rather than know one given library.

So good questions to ask might be:

1. What are the benefits of multi-threading applications?

2. What are the challenges of multi-threading applications?

3. What are some basic mechanisms for ensuring correct execution of a multi-threaded application?

4. What is 'deadlock' and can you describe any specific mechanisms for avoiding it?

5. What is a spin-lock and when (if ever) are they appropriate? What are the alternatives and when (if ever) are they appropriate?

6. What is the best way to establish the correctness of locking strategies in a program?

7. Tell me about a time when you encountered a synchronisation problem in a multi-threaded program and how it got fixed?

If they only say 'testing' to no. 6 they automatically fail the interview. If they say 'it's never happened' to no. 7 but claim experience they're a liar and automatically fail the interview.

2.2k Views · View Upvoters

⌃ Upvote · 1　　🔄 Share　　　　　　　　　　　　⌄　⤳　⚬⚬⚬

　　　　　Add a comment...　　　　　　　　　　Recommended　All

Drew Eckhardt, programming over 30 years including over 20 professionally
Updated Mar 17, 2017

Simple ones which require semaphores or condition variables for acceptable real world performance. Ones involving thread pools. Ones where assumptions involving execution order like which thread acquires a lock can be wrong.

Questions like those on other areas of programming where the goal is filtering out people who just don't get it, not identifying rare exceptional individuals. Questions which people who memorize facts but can't apply them won't answer

Related Questions

What are some good concurrency /multi-threading/ parallel programming questions for white board interview?

How can I efficiently debug multi-threaded C/C++ programs on Linux?

What are the sites for Java threading programming practice and thread programming interview questions?

What is the best approach to become more proficient at C++?

What are some good best resources to prepare C++ multi-threading scenario based interview questions for 2+ experience?
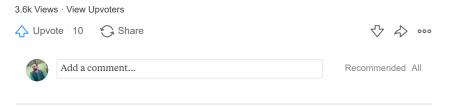
What are the best books for multi-threaded programming in C?

How did concurrent and multi-threaded C++ programming look like before C++11?

What are some interesting C program interview questions?

＋ Ask New Question

correctly. Questions which competent people will answer in a few minutes after which the conversation can proceed to more involved and interesting subjects.

Lots of people who claim to do threaded programming spin (optionally sleeping periodically, typically for whole seconds) on shared resources (like queued operations) becoming available.

Many people remember semaphores from college, but don't use condition variables (or java monitors) which work better in the real world. They don't replicate a count like queue depth in a separate object which could get out of sync. Using them means the authoritative state is in program memory which will be persisted in crash dumps. They communicate intent more clearly when you're waiting for multiple things (shutdown requested, no new requests queued, and no executing requests) or one of several things (read requests or write requests).

Many of those people cannot use a condition variable when its semantics are explained.

You shouldn't find specific examples on Quora because people reuse their interview questions so they know they're reasonable, how long a variety of engineers take to solve them, and the possible approaches so they can coach candidates who understand what they're doing but are slow in the interview environment. Having your questions where they come up in a pre-interview google search would be bad. Sharing other peoples' questions is unethical.

Careercup has a web page of questions with varying quality from people who don't understand that or don't play nice with others:

Threads Interview Questions

3.1k Views · View Upvoters

⬆ Upvote · 6   ↻ Share   ⬇ ↱ ooo

Add a comment...   Recommended All

Related Questions

What are some good concurrency /multi-threading/ parallel programming questions for white board interview?

How can I efficiently debug multi-threaded C/C++ programs on Linux?

What are the sites for Java threading programming practice and thread programming interview questions?

What is the best approach to become more proficient at C++?

What are some good best resources to prepare C++ multi-threading scenario based interview questions for 2+ experience?

What are the best books for multi-threaded programming in C?

How did concurrent and multi-threaded C++ programming look like before C++11?

What are some interesting C program interview questions?

+ Ask New Question

Top Stories from Your Feed