Unix Threads Job Interview Questions And Answers

About Interview Questions Answers

Interview Questions Answers . ORG is an interview preparation guide of thousands of Job Interview Questions And Answers, Job Interviews are always stressful even for job seekers who have gone on countless interviews. The best way to reduce the stress is to be prepared for your job interview. Take the time to review the standard interview questions you will most likely be asked. These interview questions and answers on Unix Threads will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts.

If you find any **question or answer** is incorrect or incomplete then you can **submit your question or answer** directly with out any registration or login at our website. You just need to visit Unix Threads Interview Questions And Answers to add your answer click on the *Submit Your Answer* links on the website; with each question to post your answer, if you want to ask any question then you will have a link *Submit Your Question*; that's will add your question in Unix Threads category. To ensure quality, each submission is checked by our team, before it becomes live. This Unix Threads Interview preparation PDF was generated at **Wednesday 29th November, 2023**

You can follow us on FaceBook for latest Jobs, Updates and other interviews material. www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter for latest Jobs and interview preparation guides. https://twitter.com/InterviewQA

If you need any further assistance or have queries regarding this document or its material or any of other inquiry, please do not hesitate to contact us.

Best Of Luck.

Interview Questions Answers.ORG Team
https://InterviewQuestionsAnswers.ORG/
Support@InterviewQuestionsAnswers.ORG

Unix Threads Interview Questions And Answers Guide.

Question - 1:

Explain What are threads?

Ans:

A thread is an encapsulation of the flow of control in a program. Most people are used to writing single-threaded programs - that is, programs that only execute one path through their code "at a time". Multithreaded programs may have several threads running through different code paths "simultaneously".

Question - 2:

Explain What are the main families of threads?

Ans:

There are two main families of threads:
POSIX-style threads, which generally run on Unix systems.
Microsoft-style threads, which generally run on PCs.
These families can be further subdivided.

Question - 3:

Explain Critical section?

Ans:

Necessary and sufficient conditions for a solution to the c.s. problem:
1. Mutual Exclusion --- if is executing in one of its critical sections, no , , is executing in its critical sections.
2. Progress --- a process operating outside of its critical section cannot prevent other processes from entering theirs; processes attempting to enter their critical sections simultaneously must decide which process enters eventually.
3. Bounded Waiting --- a process attempting to enter its critical region will be able to do so eventually.
Assumptions:
1. No assumptions made about relative speed of processes
2. No process may remain in its critical section indefinitely (may not terminate in its critical section)
3. A memory operation (read or write) is atomic --- cannot be interrupted. For now, we do not assume indivisible RMW cycles.

Question - 4:

What is Protection boundary?

Ans:

A protection boundary protects one software subsystem on a computer from another, in such a way that only data that is explicitly shared across such a boundary is accessible to the entities on both sides. In general, all code within a

protection boundary will have access to all data within
that boundary.
The canonical example of a protection boundary on most
modern systems is that between processes and the kernel.
The kernel is protected from processes, so that they can
only examine or change its internal state in certain
strictly-defined ways.
Protection boundaries also exist between individual
processes on most modern systems. This prevents one buggy
or malicious process from wreaking havoc on others.

Question - 5:

What is Scheduling?

Ans:

Scheduling is a key concept in computer multitasking,
multiprocessing operating system and real-time operating
system designs. Scheduling refers to the way processes are
assigned to run on the available CPUs, since there are
typically many more processes running than there are
available CPUs. This assignment is carried out by softwares
known as a scheduler and dispatcher.
The scheduler is concerned mainly with:
   * Throughput - number of processes that complete their
execution per time unit.
   * Latency, specifically

Question - 6:

What are the different kinds of threads?

Ans:

Only two typs of threreads,that are user space and kernel
supported threads.The user space threads can run only one
machine.where as the kernel supported can run on two or
more machincs simulatinously.

Question - 7:

What are the Architectural differences between User-space threads, and Kernel-supported threads?

Ans:

User-space threads live without any support from the
kernel; they maintain all of their state in user space.
Since the kernel does not know about them, they cannot be
scheduled to run on multiple processors in parallel.
Kernel-supported threads fall into two classes.
In a "pure" kernel-supported system, the kernel is
responsible for scheduling all threads.
Systems in which the kernel cooperates with a user-level
library to do scheduling are known as two-level, or hybrid,
systems. Typically, the kernel schedules LWPs, and the user-
level library schedules threads onto LWPs.
Because of its performance problems (caused by the need to
cross the user/kernel protection boundary twice for every
thread context switch), the former class has fewer members
than does the latter (at least on Unix variants). Both
classes allow threads to be run across multiple processors
in parallel.

Question - 8:

What are the Performance differences between User-space threads and Kernel-supported threads.?

Ans:

In terms of context switch time, user-space threads are the
fastest, with two-level threads coming next (all other
things being equal). However, if you have a multiprocessor,
user-level threads can only be run on a single CPU, while
both two-level and pure kernel-supported threads can be run
on multiple CPUs simultaneously.

Question - 9:

When should we use thread-safe "_r" library calls?

Ans:

If your system provides threads, it will probably provide a
set of thread-safe variants of standard C library routines.
A small number of these are mandated by the POSIX standard,
and many Unix vendors provide their own useful supersets,
including functions such as gethostbyname_r().
Unfortunately, the supersets that different vendors support
do not necessarily overlap, so you can only safely use the
standard POSIX-mandated functions. The thread-safe routines
are conceptually "cleaner" than their stateful
counterparts, though, so it is good practice to use them
wherever and whenever you can.

View All Answers

Question - 10:

How to work UNIX commands on Windows XP without installing UNIX O/S in PC?

Ans:

Install virtual machine for running unix command that may
be available from different different vendor like Ex:-
vmware player,etc

View All Answers

Question - 11:

Explain similarities between thread and process?

Ans:

- each process must have at least one thread running within it, and each thread must be running within a process.
- each process gets it own address space and memory allocation by OS, where as thread makes use of its parent process resources.
- when parent process dies, all its child process dies, but v-v is not true

View All Answers

Question - 12:

How to edit the network interface device type in container(zone) in Solaris 10?

Ans:

just edit
vi /etc/zones/<zonename.xml> file
then change the value of physical=<nic>
and save it.
after that reboot zone.
by
zoneadm -z <zonename> boot
zlogin <zonename>

View All Answers

OS Unix Most Popular & Related Interview Guides

1 : **Unix IPC Interview Questions and Answers.**

2 : **Unix General Interview Questions and Answers.**

3 : **Unix System Calls Interview Questions and Answers.**

Follow us on FaceBook
www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter
https://twitter.com/InterviewQA

For any inquiry please do not hesitate to contact us.

Interview Questions Answers.ORG Team
https://InterviewQuestionsAnswers.ORG/
support@InterviewQuestionsAnswers.ORG