

Unix Inter-Process Communication (IPC) Interview Questions And Answers Guide.



Global Guideline.

<https://www.globalguideline.com/>



Unix Inter-Process Communication (IPC) Job Interview Preparation Guide.

Question # 1

How to get or set an environment variable from a program?

Answer:-

to get environment variable execute `getenv()`;
to set environment variable execute `setenv()`;

[Read More Answers.](#)

Question # 2

What are the system calls used for process management?

Answer:-

This should not allow to increase process priority.

[Read More Answers.](#)

Question # 3

Predict the output of the following program code

```
main()
{
    fork();
    printf("Hello World!");
}
```

Answer:-

prints Hello world Hello world

All the statements after the call to `fork()` will be executed twice

[Read More Answers.](#)

Question # 4

Please Explain `fork()` system call?

Answer:-

`fork()` is a system call used for creating child processes of a parent process. it returns the process id of the created child process. after that `pid` (process id) is checked if it is negative, it means no child process is created, `pid==0` implies the id of the newly created process and `pid>0` is the id of child process given to the parent process. the statements following `fork` system call are executed by both the parent and child process. and one more thing, the parent and child process have the exact copy of address space but they exist separately for the two processes.

[Read More Answers.](#)

Question # 5

What are various IDs associated with a process?

Answer:-

Real User id, Effective User Id, saved user id.
Real Group id, Effective Group Id, saved Group id.



[Read More Answers.](#)

Question # 6

Explain the initial process sequence while the system boots up?

Answer:-

While booting, special process called the 'swapper' or 'scheduler' is created by the Process-ID 0. The swapper manages memory allocation for processes and influences CPU allocation. The swapper in turn creates 3 children: the process dispatcher, vhand, etc...

[Read More Answers.](#)

Question # 7

Tell me set-user-id is related to (in unix)?

Answer:-

setuid short for set user ID upon execution is Unix access rights flag that allow users to run an executable with the permissions of the executable's owner. This is often used to allow users on a computer system to run programs with temporarily elevated privileges in order to perform a specific task. While the assumed user id privileges provided are not always elevated, at a minimum that is specific. setuid and setgid are needed for tasks that require higher privileges than those which a common user has, such as changing his or her login password. Some of the tasks that require elevated privileges may not immediately be obvious, though - such as the ping command, which must send and listen for control packets on a network interface.

[Read More Answers.](#)

Question # 8

What is the condition required for dead lock in Unix system?

Answer:-

deadlock occurs because of

- 1.no preemption
- 2.circular waiting i.e.. when A process is waiting for the resources which are held by the process B, which is waiting for the resources engaged by process A.

[Read More Answers.](#)

Question # 9

What is i-node numbers?

Answer:-

File is identified by the I-node number in linux. Kernel uses inode number to access the file. It is unique for the particular filesystem. Inode contains all the information about the file like , file size, access permissions, time stamp , uid , gid , pointers to data blocks..etc.

[Read More Answers.](#)

Question # 10

Max relax-able permission value with out giving write permission to others?

Answer:-

```
chmod 755 <filename>
```

[Read More Answers.](#)

Question # 11

What is fork()?

Answer:-

Fork system call is used to create a new process. Fork is called once and returns twice. It will return 0 to the newly created process (child process) and process id of child to the calling process (parent process).

The child process gets copy of parents data, stack and heap segment. The code segment is common for both. Both the processes will resume execution from the command next to fork. Scheduler decide which process to run first.



[Read More Answers.](#)

Question # 12

What is ln(linking)?

Answer:-

Linking is also two types.

1>Hard link.

2>soft link.

For hard :-

ln <source> <destination>

for softlink :-

ln -s <source> <destination>

[Read More Answers.](#)

Question # 13

Explain Linking across directories?

Answer:-

It's possible through both symbolic and hard links. However, hard links are only possible if both source and destination belong to the same volume.

[Read More Answers.](#)

Question # 14

What is the very first process created by kernel?

Answer:-

The first process in Unix is INIT.

[Read More Answers.](#)

Question # 15

What is the process id for kernel process?

Answer:-

zero(0)

[Read More Answers.](#)

Question # 16

Explain What is the process id for kernel process?

Answer:-

The Process id are sequentially numbered and the first process id created is 'PID 0', which loads the data structures and resources of Kernel which also forks the next process "INIT" which has the process id '1'.

[Read More Answers.](#)

Question # 17

What is semaphore?

Answer:-

a semaphore is nothing but a term used in unix for a variable which acts as a counter. for instance there may be times when two processes try to access the same file simultaneously. in this event we must control the access of the when the other process is accessing this is done by assigning value to a semaphore. the value of the semaphore is initialized by the 1st process when the file is in access by it. when the 2nd process try to access the file it checks the value of the semaphore and if it finds the value as initialized it does not access the file.

[Read More Answers.](#)

Question # 18

How to write the program on full-duplex communication on bidirectional(e.g using two pipes)?

Answer:-

```
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<sys/ipc.h>
#include<sys/types.h>
int main()
```



```
{
int p1[2],p2[2],i;
char buf[5];
pipe(p1);
pipe(p2);
if(fork()==0)
{
printf("
this is child(the input text is text)
");
close(p1[0]);
close(p2[1]);
write(p1[1],"text",5);
read(p2[0],buf,5);
write(1,buf,5);
}
else
{
printf("
this is parent(the output text is text)
");
close(p2[0]);
close(p1[1]);
read(p1[0],buf,5)
for(i=0;i<4;i++)
buf[i]=to upper(buf[i]);
write(p2[1],buf,5)
}
}
```

[Read More Answers.](#)

Question # 19

What are the system calls used for process management:

Answer:-

The system calls for process management are
Fork() > create child process
Vfork() > create child process (copy-on-write only)
exec() > exec to do a different task
wait() > wait for a process to complete execution
kill() > to send a signal to a process
signal() > to handle a signal
sigaction() > handle signal
exit() > exit from a process execution
_exit() > same as exit() but with out clean up
and many more

[Read More Answers.](#)

Question # 20

Please Describe the initial process sequence while the system boots up?

Answer:-

â€"Hardware Boot
â€"Read boot block (block 0) and load to memory
â€"Transfer control to kernel
â€"Kernel initialization
â€"Mount the root file system on / and create environment for process 0
â€"Process 0 invokes process 1 (init process)
â€"init process invokes process getty
â€"getty invokes login

[Read More Answers.](#)

Question # 21

What is IPC? What are the various schemes available?

Answer:-

Inter Process Communication. IPC is used to pass information between two or more processes.
Schemes are pipes, shared memory & semaphore.
Below are the different IPC methods
1. Semaphores
2. FIFO's (Also called Named Pipes)
3. Message Queues
4. Shared Memory

[Read More Answers.](#)

Question # 22



How do you execute one program from within another?

Answer:-

by calling system call fork() to create child process which handles the other.

[Read More Answers.](#)

Question # 23

What is an advantage of executing a process in background?

Answer:-

There r two main advantage :

- 1.you can use console to execute your command.
- 2.More importantly if one is connected through remote console and as there is no controlling terminal attached to background process even if your terminal get disconnected the process continues (with help of nohup in shell script or by using setsid in C)

[Read More Answers.](#)

Question # 24

What Happens when we execute a Unix command?

Answer:-

When command is given then unix os will fork the shell i.e will creat a new process and will execute the command using exec command...something like suppose you gave command "ls" in the shell then...

```
fork();
```

```
exec(ls);
```

It will give you the result and after this the child process will die.

[Read More Answers.](#)

Question # 25

What is Daemon?

Answer:-

The processes like vhand, bdfush, sched are housed in kernel file or /unix system which are known as daemons. These files run in the background without users request. These are created when the system boots up and remains active till it shut down or hang. These are not linked to any user or any terminal .We can't kill a daemon.

[Read More Answers.](#)

Question # 26

Explain a zombie?

Answer:-

A system that has been taken over using Remote Control Software. Zombies are often used to send spam or to attack remote servers with an overwhelming amount of traffic (a Distributed Denial of Service Attack).

[Read More Answers.](#)

OS Unix Most Popular Interview Topics.

- 1 : [Unix Threads Frequently Asked Interview Questions and Answers Guide.](#)
- 2 : [Unix General Frequently Asked Interview Questions and Answers Guide.](#)
- 3 : [Unix System Calls Frequently Asked Interview Questions and Answers Guide.](#)

About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of [Interview Questions with Answers](#) or begin a tutorial right away, such as [HTML](#), [XML](#), [XSLT](#), [Cascading Style Sheet \(CSS\)](#), [Search Engine Optimization \(SEO\)](#), [JavaScript](#), [Structure Query Language \(SQL\)](#), [Database Articles](#), [Web Hosting Guide](#) and much more. Learn the most common technologies [Interview Questions and Answers](#). We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from <https://www.GlobalGuideline.com> at **April 4th, 2021**

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and [e-mail us](#) we will fix it.

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

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

Best Of Luck.

Global Guideline Team
<https://www.globalguideline.com>
Support@globalguideline.com