Linux Debugging Job Interview Questions And Answers

Interview Questions 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 Linux Debugging 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 Linux Debugging 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 Linux Debugging category. To ensure quality, each submission is checked by our team, before it becomes live. This Linux Debugging 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

Linux Debugging Interview Questions And Answers Guide.

Question - 1:

Which one of the following command saves the command history of GDB in a file?
a) history
b) set history
c) set history save on
d) none of the mentioned

Ans:

c) set history save on

View All Answers

Question - 2:

In GDB, a trace-point can be set by the command
a) trace
b) set
c) break trace
d) none of the mentioned

Ans:

a) trace

View All Answers

Question - 3:

The command "show commands" of GDB
a) displays the last 10 commands in the command history
b) displays all commands of the command history
c) displays all the commands available in GDB
d) none of the mentioned

Ans:

a) displays the last 10 commands in the command history

View All Answers

Question - 4:

The GDB command "show output-radix"
a) sets the default base for numeric display
b) displays the current default base for numeric display
c) both (a) and (b)
d) none of the mentioned

Ans:

b) displays the current default base for numeric display

View All Answers

Question - 5:

The user can define a command for GDB with the command:
a) define
b) command
c) assign
d) none of the mentioned

Ans:

a) define

View All Answers

Question - 6:

By default the GDB automatically executes the command from its:
a) init files
b) start files
c) begin files
d) none of the mentioned

Ans:

a) init files

Question - 7:

The GDB text user interface uses the _____ library to show the source file.
a) curses
b) YUI
c) JUI
d) none of the mentioned

Ans:

a) curses

Question - 8:

Which one of the following GDB command allows to move from one stack frame to another without printing the frame?
a) select-frame
b) frame
c) frame move
d) none of the mentioned

Ans:

a) select-frame

Question - 9:

Which one of the following GDB command deletes any break-point at the next instruction to be executed in the selected stack frame?
a) clear
b) delete
c) disable
d) none of the mentioned

Ans:

a) clear

Question - 10:

The result of an expression can be assigned to an environment variable with the command:
a) assign
b) set
c) env
d) none of the mentioned

Ans:

b) set

Question - 11:

In GDB hardware-dependent information about the floating point unit can be displayed by the command
a) info float
b) display float
c) show float
d) none of the mentioned

Ans:

a) info float

Question - 12:

In GDB, we can refer to machine register contents, in expressions, as variables with names starting with:
a) $
b) #
c) !
d) none of the mentioned

Ans:

a) $

Question - 13:

Which one of the following variables is used within GDB to hold on to a value and refer to it later?
a) convenience variables
b) environment variables
c) temporary variables
d) none of the mentioned

Ans:

a) convenience variables

Question - 14:

If we want to print the value of a variable in hexadecimal, we have to use "print" command with the option ___ in GDB.
a) x
b) h
c) hex
d) none of the mentioned

Ans:

a) x

Question - 15:

In GDB which one of the following allows us to specify a variable in terms of the file or function where it is defined?
a) ::
b) @
b) $
d) none of the mentioned

Ans:

a) ::

Question - 16:

With the list command, by default GDB prints the _____ source lines.
a) 20
b) 10
c) all
d) none of the mentioned

Ans:

b) 10

Question - 17:

Which one of the following is a special breakpoint that stops the program when the value of an expression changes in GDB?
a) watchpoint
b) catchpoint
c) getpoint
d) none of the mentioned

Ans:

a) watchpoint

Question - 18:

Inside GDB, a program may stop because of
a) a signal
b) a breakpoint
c) step command
d) all of the mentioned

Ans:

d) all of the mentioned

Question - 19:

While debugging with GDB, arguments to the program can be specified by the arguments of _____ command.
a) run
b) gdb

c) make
d) none of the mentioned

Ans:

a) run

Question - 20:

What is the output of this program no 16?
```
#include<stdio.h>
#include<sys/types.h>
#include<netinet/in.h>
#include<sys/socket.h>
#include<errno.h>

int main()
{
    struct sockaddr_in addr;
    int fd;
    fd = socket(AF_UNIX,SOCK_STREAM,0);
    if (fd == -1)
        perror("socket");
    addr.sun_family = AF_UNIX;
    strcpy(addr.sun_path,"san_sock");
    if (bind(4,(struct sockaddr*)&addr,sizeof(addr)) == -1)
        printf("Sanfoudnryn");
    return 0;
}
```
a) error
b) "google"
c) segmentation fault
d) none of the mentioned

Ans:

a) error
Explanation:
The structure used for AF_UNIX if sockaddr_un.
Output:
[root@localhost google]# gcc -o san san.c
san.c: In function 'main':
san.c:14:6: error: 'struct sockaddr_in' has no member named 'sun_family'
san.c:15:2: warning: incompatible implicit declaration of built-in function 'strcpy' [enabled by default]
san.c:15:13: error: 'struct sockaddr_in' has no member named 'sun_path'
[root@localhost google]#

Question - 21:

What is the output of this program no 15?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>

int main()
{
    struct sockaddr_un add_server, add_client;
    int fd_server, fd_client;
    int len;
    char ch;
    fd_server = socket(AF_UNIX,SOCK_STREAM,0);
    if(fd_server == -1)
        perror("socket");
    add_server.sun_family = AF_UNIX;
    strcpy(add_server.sun_path,"san_sock");
    if( bind(fd_server,(struct sockaddr*)&add_server,sizeof(add_server)) != 0)
        perror("bind");
    len = sizeof(add_client);
    fd_client = connect(fd_server,(struct sockaddr*)&add_client,&len);
    printf("googlen");
    return 0;
}
```
a) this program will print the string "google"
b) segmentation fault
c) error
d) none of the mentioned

Ans:

c) error
Explanation:
The syntax of the connect() is wrong. connect() should be used in client program only.

Ouptut:
[root@localhost google]# gcc -o san san.c
san.c: In function 'main':
san.c:20:46: warning: passing argument 3 of 'connect' makes integer from pointer without a cast [enabled by default]
/usr/include/sys/socket.h:129:12: note: expected 'socklen_t' but argument is of type 'int *'
[root@localhost google]#

View All Answers

Question - 22:

What this program is not able to connect with any client program?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>

int main()
{
    struct sockaddr_un add_server, add_client;
    int fd_server, fd_client;
    int len;
    char ch;
    fd_server = socket(AF_UNIX,SOCK_STREAM,0);
    if(fd_server == -1)
        perror("socket");
    add_server.sun_family = AF_UNIX;
    strcpy(add_server.sun_path,"san_sock");
    if( bind(fd_server,(struct sockaddr*)&add_server,sizeof(add_server)) != 0)
        perror("bind");
    len = sizeof(add_client);
    fd_client = accept(fd_server,(struct sockaddr*)&add_client,&len);
    printf("googlen");
    return 0;
}
```
a) the listen() is missing
b) the connect() is missing
c) the read() and write() are missing
d) none of the mentioned

Ans:

a) the listen() is missing

View All Answers

Question - 23:

What is the output of this program no 14?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>
#include<errno.h>

int main()
{
    struct sockaddr_un addr;
    int fd;
    fd = socket(AF_UNIX,SOCK_STREAM,0);
    if (fd == -1)
        perror("socket");
    addr.sun_family = AF_UNIX;
    strcpy(addr.sun_path,"san_sock");
    if (bind(4,(struct sockaddr*)&addr,sizeof(addr)) == -1)
        printf("Sanfoudnryn");
    return 0;
}
```
a) this program will print the string "google"
b) this program will not print the string "google"
c) segmentation fault
d) none of the mentioned

Ans:

a) this program will print the string "google"
Explanation:
The first argument of the bind() is not a valid file descriptor in this program.
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
Sanfoudnry
[root@localhost google]#

View All Answers

Question - 24:

What is the output of this program?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>

int main()
{
    int fd;
    fd = socket(AF_UNIX,SOCK_STREAM,0);
    printf("%dn",fd);
    return 0;
}
```
a) 0
b) 1
c) 2
d) 3

Ans:

d) 3
Explanation:
The socket() returns the lowest available file descriptor and in this program i.e. 3.
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
3
[root@localhost google]#

View All Answers

Question - 25:

What is the output of the program no 13?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>

int main()
{
    struct sockaddr_un add_server, add_client;
    int fd_server, fd_client;
    int len;
    char ch;
    fd_server = socket(AF_UNIX,SOCK_STREAM,0);
    if(fd_server == -1)
        perror("socket");
    add_server.sun_family = AF_UNIX;
    strcpy(add_server.sun_path,"san_sock");
    if( bind(fd_server,(struct sockaddr*)&add_server,sizeof(add_server)) != 0)
        perror("bind");
    if( listen(fd_server,3) != 0)
        perror("listen");
    len = sizeof(add_client);
    fd_client = accept(fd_server,(struct sockaddr*)&add_client,&len);
    printf("googlen");
    return 0;
}
```
a) the program will print the string "google"
b) the process will remain block
c) segmentation fault
d) none of the mentioned

Ans:

b) the process will remain block
Explanation:
There is no peding request in the queue for listening socket "san_sock".
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
^Z
[4]+ Stopped ./san
[root@localhost google]#

View All Answers

Question - 26:

What is the length of of the queue for pending connections in this program?
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>
```

```
int main()
{
    struct sockaddr_un add_server;
    int fd_server;
    fd_server = socket(AF_UNIX,SOCK_STREAM,0);
    if(fd_server == -1)
        perror("socket");
    add_server.sun_family = AF_UNIX;
    strcpy(add_server.sun_path,"server_sock2");
    if( bind(fd_server,(struct sockaddr*)&add_server,sizeof(add_server)) != 0)
        perror("bind");
    if( listen(fd_server,3) != 0)
        perror("listen");
    return 0;
}
```
a) 0
b) 1
c) 2
d) 3

Ans:

d) 3
Explanation:
The second argument of listen() specifies the length for the queue for pending connections.

Question - 27:

By this program the soket "san_sock" will create
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/un.h>
#include<sys/socket.h>

int main()
{
    struct sockaddr_un add_server;
    int fd_server;
    fd_server = socket(AF_UNIX,SOCK_STREAM,0);
    if(fd_server == -1)
        perror("socket");
    add_server.sun_family = AF_UNIX;
    strcpy(add_server.sun_path,"san_sock");
    if( bind(fd_server,(struct sockaddr*)&add_server,sizeof(add_server)) != 0)
        perror("bind");
    return 0;
}
```
a) in the /tmp directory
b) in the /usr directory
c) in the present working directory
d) none of the mentioned

Ans:

c) in the present working directory
Output:
[root@localhost google]# ls
san.c
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
[root@localhost google]# ls
san san.c san_sock
[root@localhost google]#

Question - 28:

In this program, the third argument of the socket() is used for _____ potocol.
```
#include<stdio.h>
#include<sys/types.h>
#include<sys/socket.h>
int main()
{
    int fd_socket;
    if(socket(AF_UNIX,SOCK_STREAM,0) == -1)
        perror("socket");
    return 0;
}
```
a) TCP/IP
b) UDP
c) both (a) and (b)
d) none of mentioned

Ans:
a) TCP/IP

Question - 29:

What is the output of this program no 12?
```c
#include<stdio.h>

int main()
{
    int fd_socket;
    fd_socket = socket(AF_UNIX,SOCK_STREAM,0);
    printf("%dn",fd_socket);
    return 0;
}
```
a) -1
b) 0
c) any integer value
d) none of the mentioned

Ans:

d) none of the mentioned
Explanation:
To use socket(), the header files sys/types.h and sys/socket.h are required.
Output:
[root@localhost google]# gcc -o san san.c
san.c: In function 'main':
san.c:6:21: error: 'AF_UNIX' undeclared (first use in this function)
san.c:6:21: note: each undeclared identifier is reported only once for each function it appears in
san.c:6:29: error: 'SOCK_STREAM' undeclared (first use in this function)
[root@localhost google]#

Question - 30:

What is the the response of this server for this client if both programs are running on the same system?
```c
/*This is server.c*/
#include<stdio.h>
#include<stdlib.h>
#include<netinet/in.h>
#include<sys/types.h>
#include<sys/socket.h>

int main()
{
    int fd_server, fd_client, len, len_client;
    struct sockaddr_in add_server, add_client;
    char buff[10];
    fd_server = socket(AF_INET,SOCK_STREAM,0);
    if (fd_server == -1){
        perror("fd_sock");
        exit(1);
}
add_server.sin_family = AF_INET;
add_server.sin_port = htons(4001);
add_server.sin_addr.s_addr = inet_addr("127.0.0.1");
len = sizeof(add_server);
len = sizeof(add_client);
if( bind(fd_server,(struct sockaddr*)&add_server,len) != 0)
    perror("bind");
    if(listen(fd_server,5) != 0)
        perror("listen");
    fd_client = accept(fd_server,(struct sockaddr*)&add_client,&len_client);
    if(fd_client == -1)
        perror("accept");
    read(fd_client,buff,10);
    return 0;
}
/*This is the client.c*/
#include<stdio.h>
#include<netinet/in.h>
#include<sys/types.h>
#include<sys/socket.h>

int main()
{
    int fd_client,fd, len;
    struct sockaddr_in add_server;
    fd_client = socket(AF_INET,SOCK_STREAM,0);
    if (fd_client == -1){
        perror("fd_sock");
```

```
        exit(1);
   }
   add_server.sin_family = AF_INET;
   add_server.sin_port = ntohs(4001);
   add_server.sin_addr.s_addr = inet_addr("127.0.0.1");
   len = sizeof(add_server);
   fd  = connect(fd_client,(struct sockaddr*)&add_server,len);
   if(fd == -1)
       perror("connect");
       write(fd,"Hellon",6);
       return 0;
   }
```
a) the server will write back to the client whatever the clinet will write to the server
b) the client server communication will not work
c) the response can not be determined
d) none of the mentioned

Ans:

a) the server will write back to the client whatever the client will write to the server
Explanation:
The loopback address is used as IP address in both the programs.

View All Answers

Question - 31:

On which system call, this program (process) waits until the server responds?
```
   #include<stdio.h>
   #include<netinet/in.h>
   #include<sys/types.h>
   #include<sys/socket.h>

   int main()
   {
       int fd_client,fd, len;
       struct sockaddr_in add_server;
       fd_client = socket(AF_INET,SOCK_STREAM,0);
       if (fd_client == -1){
           perror("fd_sock");
           exit(1);
   }
   add_server.sin_family = AF_INET;
   add_server.sin_port = ntohs(4001);
   add_server.sin_addr.s_addr = inet_addr("127.0.0.1");
   len = sizeof(add_server);
   fd  = connect(fd_client,(struct sockaddr*)&add_server,len);
   if(fd == -1)
       perror("connect");
       write(fd,"Hellon",6);
       return 0;
   }
```
a) socket()
b) connect()
c) both (a) and (b)
d) none of the mentioned

Ans:

a) socket()

View All Answers

Question - 32:

What is the output of this program no 11?
```
   #include<stdio.h>
   #include<stdlib.h>
   #include<netinet/in.h>
   #include<sys/types.h>
   #include<sys/socket.h>

   int main()
   {
       int fd_server, fd_client, len, len_client;
       struct sockaddr_in add_server;
       fd_server = socket(AF_INET,SOCK_STREAM,0);
       close(fd_server);
           perror("accept");
       if(listen(fd_server,5) != 0)
           perror("listen");
       fd_client = accept(fd_server,(struct sockaddr*)&add_server,&len);
           if(fd_client == -1)
       return 0;
   }
```
a) syntax error

b) error at the time of compilation
c) segmentation fault
d) none of the mentioned

Ans:

d) none of the mentioned
Explanation:
The program will not work properly because the file descriptor is not available in the for listen() and accept().
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
accept: Success
listen: Bad file descriptor
[root@localhost google]#

<u>View All Answers</u>

Question - 33:

What is the output of this program no 10?
```
  #include<stdio.h>
  #include<stdlib.h>
  #include<netinet/in.h>
  #include<sys/types.h>
  #include<sys/socket.h>

  int main()
  {
     int fd_server, fd_client, len, len_client;
     struct sockaddr_in add_server;
     fd_server = socket(AF_INET,SOCK_STREAM,0);
     fd_client = accept(fd_server,(struct sockaddr*)&add_server,&len);
     if(fd_client == -1)
        perror("accept");
     if(listen(fd_server,5) != 0)
        perror("listen");
     return 0;
  }
```
a) syntax error
b) error at the time of compilation
c) segmentation fault
d) none of the mentioned

Ans:

d) none of the mentioned
Explanation:
The listen() must always be used before accept().
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
accept: Invalid argument
[root@localhost google]#

<u>View All Answers</u>

Question - 34:

What is the problem with this server program?
```
  #include<stdio.h>
  #include<stdlib.h>
  #include<netinet/in.h>
  #include<sys/types.h>
  #include<sys/socket.h>

  int main()
  {
     int fd_server, fd_client, len;
     struct sockaddr_in add_server;
     fd_server = socket(AF_INET,SOCK_STREAM,0);
     if (fd_server == -1){
        perror("fd_sock");
        exit(1);
     }
     add_server.sin_family = AF_INET;
     add_server.sin_port = htons(4001);
     add_server.sin_addr.s_addr = inet_addr("122.23.1.1");
     len = sizeof(add_server);
     if( bind(fd_server,(struct sockaddr*)&add_server,len) != 0)
        perror("bind");
     if(listen(fd_server,5) != 0)
        perror("listen");
     fd_client = accept(fd_server,(struct sockaddr*)&add_server,&len);
     if(fd_client == -1)
        perror("accept");
```

```
   return 0;
 }
```
a) it can not accept the request of any client
b) it will give the segmentation fault
c) there is no problem with this program
d) none of the mentioned

Ans:

a) it can not accept the request of any client

Question - 35:

What is the output of this program?
```
 #include<stdio.h>
 #include<sys/socket.h>
 int main()
 {
    int ret;
    ret = shutdown(0,0);
    printf("%dn",ret);
    return 0;
 }
```
a) 0
b) -1
c) can not be determined
d) none of the mentioned

Ans:

b) -1
Explanation:
The shutdown() is used to close a socket and the first argument in shutdown() is socket.
Output:
[root@localhost google]# gcc -o san san.c
[root@localhost google]# ./san
-1
[root@localhost google]#

Question - 36:

This program is valid for
```
 #include<stdio.h>
 #include<netinet/in.h>
 #include<sys/types.h>
 #include<sys/socket.h>

 int main()
 {
    int fd_client,fd, len;
    struct sockaddr_in add_server;
    fd_client = socket(AF_INET,SOCK_STREAM,0);
    if (fd_client == -1){
       perror("fd_sock");
       exit(1);
    }
    add_server.sin_family = AF_INET;
    add_server.sin_port = ntohs(4001);
    add_server.sin_addr.s_addr = inet_addr("144.29.8.2");
    len = sizeof(add_server);
    fd  = connect(fd_client,(struct sockaddr*)&add_server,len);
    return 0;
 }
```
a) IPv4
b) IPv6
c) both (a) and (b)
d) none of the mentioned

Ans:

a) IPv4

Question - 37:

This program can send the request to
```
 #include<stdio.h>
 #include<netinet/in.h>
 #include<sys/types.h>
 #include<sys/socket.h>

 int main()
 {
```

```
    int fd_client,fd, len;
    struct sockaddr_in add_server;
    fd_client = socket(AF_INET,SOCK_STREAM,0);
    if (fd_client == -1){
        perror("fd_sock");
        exit(1);
    }
    add_server.sin_family = AF_INET;
    add_server.sin_port = ntohs(4001);
    add_server.sin_addr.s_addr = inet_addr("193.39.0.4");
    len = sizeof(add_server);
    fd  = connect(fd_client,(struct sockaddr*)&add_server,len);
    if(fd == -1)
        perror("connect");
    return 0;
}
```
a) the system having IP address 193.39.0.4
b) any system present in the network
c) any system of the private network
d) none of the mentioned

Ans:

a) the system having IP address 193.39.0.4
Explanation:
The IP address is mentioned in the proper element of the structure sockaddr_in

View All Answers

Question - 38:

What is the output of this program?
```
  #include<stdio.h>
  #include<stdlib.h>
  #include<netinet/in.h>
  #include<sys/types.h>
  #include<sys/socket.h>

  int main()
  {
     int fd_server, fd_client, len, len_client;
     struct sockaddr_in add_server, add_client;
     char buff[10];
     fd_server = socket(AF_INET,SOCK_STREAM,0);
     if (fd_server == -1){
        perror("fd_sock");
        exit(1);
     }
     len = sizeof(add_server);
     len_client = sizeof(add_client);
     if( bind(fd_server,(struct sockaddr*)&add_server,len) != 0)
        perror("bind");
     fd_client = accept(fd_server,(struct sockaddr*)&add_client,len_client);
     if(fd_client == -1)
        perror("accept");
     read(fd_client,buff,10);
     return 0;
  }
```
a) segmentation fault
b) error at the time of compilation
c) syntax error
d) none of the mentioned

Ans:

b) error at the time of compilation
Explanation:
The third argument of the accept is the type of pointer.
Output:
[root@localhost google]# gcc -o san san.c
san.c: In function 'main':
san.c:26:39: warning: passing argument 3 of 'accept' makes pointer from integer without a cast [enabled by default]
/usr/include/sys/socket.h:214:12: note: expected 'socklen_t * __restrict__' but argument is of type 'int'
[root@localhost google]#

View All Answers

Question - 39:

What is the output of this program no 9?
```
  #include<stdio.h>
  #include<sys/types.h>
  #include<sys/socket.h>

  int main()
  {
```

```
    struct sockaddr_in addr;
    int fd;
    fd = socket(AF_INET,SOCK_STREAM,0);
    printf("%dn",fd);
    return 0;
  }
```
a) -1
b) 3
c) error
d) none of the mentioned

Ans:

c) error
Explanation:
The header file netinet/in.h is required to use the structure sockaddr_in.
Output:
[root@localhost google]# gcc -o san san.c
san.c: In function 'main':
san.c:7:21: error: storage size of 'addr' isn't known
[root@localhost google]#

View All Answers

Question - 40:

The execution of the program in GDB can be affected by:
a) arguments
b) working directory
c) environment
d) all of the mentioned

Ans:

d) all of the mentioned

View All Answers

Question - 41:

Assemble code of the program can be displayed in GDB by the command:
a) disassemble
b) assemble
c) assembly
d) none of the mentioned

Ans:

a) disassemble

View All Answers

Question - 42:

Which one of the following is not true about the GDB?
a) info register is used to see that what is in the processor registers
b) processor registers can not be accessed by GDB
c) first 32 bits of the variable can not be examined
d) none of the mentioned

Ans:

c) first 32 bits of the variable can not be examined

View All Answers

Question - 43:

Which GDB command is used to examine the memory?
a) x
b) y
c) z
d) none of the mentioned

Ans:

a) x

View All Answers

Question - 44:

In GDB breakpoints can be skipped by the command:
a) ignore
b) reject
c) skip
d) none of the mentioned

Ans:

a) ignore

Linux Debugging Interview Questions And Answers

Question - 45:

What is temporary breakpoint?
a) it stops the program once
b) it is removed after one execution of the program
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 46:

If we have multiple source files, then during the debugging with GDB:
a) breakpoint can not be set
b) break point can be set by "break" command with a filename
c) break point can be set only to makefile
d) none of the mentioned

Ans:

b) break point can be set by "break" command with a filename

Question - 47:

The GDB command "info local"
a) displays the list of local variables
b) value of local values for the current stack frame
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 48:

GDB command "frame" is used:
a) to change the stack frames
b) to check the stack frames only
c) it is not a valid command
d) none of the mentioned

Ans:

a) to change the stack frames

Question - 49:

While debugging with GDB:
a) variables can be print
b) variables can be modify
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 50:

Which one of the following is not true about GDB?
a) quit command is used to exit the GDB
b) kill command is used to stop execution in GDB
c) if the execution is stopped by kill command then it can not be started again
d) none of the mentioned

Ans:

c) if the execution is stopped by kill command then it can not be started again

Question - 51:

We can list all the break-point in GDB by the command:
a) info break
b) break all
c) both (a) and (b)

d) none of the mentioned

Ans:

a) info break

Question - 52:

To put the breakpoint at the current line _____ command can be used?
a) b
b) break
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 53:

Which GDB command can be used to put a break-point at the beginning of the program?
a) b main
b) b start
c) break
d) none of the mentioned

Ans:

a) b main

Question - 54:

GDB can be used:
a) to find out the memory leakages
b) to get the result of a particular expression in a program
c) to find the reason of segementation fault
d) all of the mentioned

Ans:

d) all of the mentioned

Question - 55:

Which GDB command reloads the debugging information?
a) file
b) reload
c) debug
d) none of the mentioned

Ans:

a) file

Question - 56:

The "step" command of GDB:
a) executes the current line of the program
b) stops the next statement to be executed
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 57:

The specific break point can be deleted by _____ command in GDB.
a) delete
b) del
c) remove
d) none of the mentioned

Ans:

a) delete

Question - 58:

Which GDB command produces a stack trace of the function calls that lead to a segmentation fault?
a) trace
b) backtrace
c) forwardtrace
d) none of the mentioned

Ans:

b) backtrace

Question - 59:

Which GDB command interrupts the program whenever the value of a variable is modified and prints the value old and new values of the variable?
a) watch
b) show
c) trace
d) none of the mentioned

Ans:

a) watch

Question - 60:

Which GDB command prints the value of a variable in hex.
a) print/x
b) print/h
c) print/e
d) none of the mentioned

Ans:

a) print/x

Question - 61:

To print the value of a variable while debugging with GDB, _____ command can be used.
a) printf
b) print
c) show
d) none of the mentioned

Ans:

b) print

Question - 62:

At the time of debugging with GDB, if we just press ENTER:
a) GDB will repeat the same command you just gave it
b) GDB will do nothing
c) GDB will exit
d) none of the mentioned

Ans:

a) GDB will repeat the same command you just gave it

Question - 63:

In GDB debugging, we can proceed to the next break-point with command:
a) next
b) continue
c) both (a) and (b)
d) none of the mentioned

Ans:

b) continue

Question - 64:

In debugging with GDB, break points can be set to:
a) any line
b) any function
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 65:

The command "gdb google"
a) will start debugging for the file "google" if the file is compiled with -g option with GCC
b) will create executable for debugging
c) will provide all errors present in the file "google"
d) none of the mentioned

Ans:

a) will start debugging for the file "google" if the file is compiled with -g option with GCC

Question - 66:

GDB can be used for:
a) c language
b) c++ language
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 67:

GDB stands for:
a) GNU debugger
b) general debugging breakpoint
c) general debugger
d) none of the mentioned

Ans:

a) GNU debugger

Question - 68:

In GDB, breakpoints can be set by the command:
a) break
b) b
c) both (a) and (b)
d) none of the mentioned

Ans:

c) both (a) and (b)

Question - 69:

For debugging with GDB, the compiled program can be run by the command
a) run
b) execute
c) ./
d) none of the mentioned

Ans:

a) run

Question - 70:

For debugging with GDB, the file "google" can be created with the command:
a) gcc -g -o google google.c
b) gcc -g google.c
c) gdb google
d) none of the mentioned

Ans:

a) gcc -g -o google google.c

Question - 71:

What is the output of this program no 8?
    #include<stdio.h>
    #include<pthread.h>

```
 void *fun_t(void *arg);
 void *fun_t(void *arg)
 {
   sleep(1);
 }
 int main()
 {
    pthread_t pt;
    void *res_t;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
      perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
      perror("pthread_join");
    printf("%sn",res_t);
    return 0;
 }
```
a) this process will pause
b) segmentation fault
c) run time error
d) none of the mentioned

Ans:

b) segmentation fault
Explanation:
This program is trying to print the return value of the thread, but pthread_exit() function is not present in the thread.
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
Segmentation fault (core dumped)
[root@localhost google]#

View All Answers

Question - 72:

What is the output of this program no 7?
```
 #include<stdio.h>
 #include<pthread.h>
 #include<fcntl.h>

 void *fun_t(void *arg);
 void *fun_t(void *arg)
 {
    pthread_exit("Bye");
    printf("googlen");
 }
 int main()
 {
    pthread_t pt;
    void *res_t;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
      perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
      perror("pthread_join");
    printf("%sn",res_t);
    return 0;
 }
```
a) google
b) Bye
c) segementation fault
d) run time error

Ans:

b) Bye
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
Bye
[root@localhost google]#

View All Answers

Question - 73:

What is the output of this program no 6?
```
 #include<stdio.h>
 #include<pthread.h>
 #include<fcntl.h>

 int fd;
 void *fun_t(void *arg);
 void *fun_t(void *arg)
 {
    char buff[10];
```

```
    int count;
    count = read(fd,buff,10);
    printf("%dn",count);
    pthread_exit("Bye");
}
int main()
{
    pthread_t pt;
    void *res_t;
    fd = open("san.c",O_RDONLY);
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
        perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
        perror("pthread_join");
    return 0;
}
```
a) 10
b) 0
c) -1
d) segmentation fault

Ans:

a) 10
Explanation:
Open file descritpors can be shares between threads of the same process
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
10
[root@localhost google]#
View All Answers

Question - 74:

Which one of the following statement is not true about this program?
```
#include<stdio.h>
#include<pthread.h>

void *fun_t(void *arg);
void *fun_t(void *arg)
{
    printf("%dn",getpid());
    pthread_exit("Bye");
}
int main()
{
    pthread_t pt;
    void *res_t;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
        perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
        perror("pthread_join");
    printf("%dn",getpid());
    return 0;
}
```
a) both printf statements will print the same value
b) both printf statements will print the different values
c) this program will print nothing
d) none of the mentioned

Ans:

a) both printf statements will print the same value
Explanation:
All the threads of the same process have same PID.
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
12981
12981
[root@localhost google]#
View All Answers

Question - 75:

What is the output of this program no 4?
```
#include<stdio.h>
#include<pthread.h>

int a;
void *fun_t(void *arg);
void *fun_t(void *arg)
{
```

```
    a = 20;
    pthread_exit("Bye");
  }
  int main()
  {
    pthread_t pt;
    void *res_t;
    a = 10;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
      perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
      perror("pthread_join");
    printf("%dn",a);
    return 0;
  }
```
a) 10
b) 20
c) segmentation fault
d) none of the mentioned

Ans:

b) 20
Explanation:
In this program the value of variable "a" is changed by the thread "fun_t".
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
20
[root@localhost google]#

Question - 76:

What is the output of this program no 3?
```
  #include<stdio.h>
  #include<pthread.h>

  int a;
  void *fun_t(void *arg);
  void *fun_t(void *arg)
  {
    printf("%dn",a);
    pthread_exit("Bye");
  }
  int main()
  {
    pthread_t pt;
    void *res_t;
    a = 10;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
      perror("pthread_create");
    if(pthread_join(pt,&res_t) != 0)
      perror("pthread_join");
    return 0;
  }
```
a) 10
b) 0
c) -1
d) none of the mentioned

Ans:

a) 10
Explanation:
Thread of the same process shares the global variables.
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
10
[root@localhost google]#

Question - 77:

What is the output of this program no 2?
```
  #include<stdio.h>
  #include<pthread.h>

  void *fun_t(void *arg);
  void *fun_t(void *arg)
  {
    printf("%dn",a);
    pthread_exit("Bye");
```

```
    }
    int main()
    {
        int a;
        pthread_t pt;
        void *res_t;
        a = 10;
        if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
            perror("pthread_create");
        if(pthread_join(pt,&res_t) != 0)
            perror("pthread_join");
        return 0;
    }
```
a) 10
b) 0
c) -1
d) none of the mentioned

Ans:

d) none of the mentioned
Explanation:
Each thread has its own stack so local variables are not shared among thread. Hence this program will give an error.
Output:
[root@localhost google]# gcc -o san san.c -lpthread
san.c: In function 'fun_t':
san.c:7:16: error: 'a' undeclared (first use in this function)
san.c:7:16: note: each undeclared identifier is reported only once for each function it appears in
[root@localhost google]#

View All Answers

Question - 78:

What is the output of this program no 1?
```
    #include<stdio.h>
    #include<pthread.h>

    void *fun_t(void *arg);
    void *fun_t(void *arg)
    {
        printf("googlen");
        pthread_exit("Bye");
    }
    int main()
    {
        pthread_t pt;
        void *res_t;
        if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
            perror("pthread_create");
        return 0;
    }
```
a) this program will print the string "google"
b) this program will print nothing
c) segmentation fault
d) run time error

Ans:

b) this program will print nothing
Explanation:The pthread_join() function waits for the thread to terminate.
Output:
[root@localhost google]# gcc -o san san.c -lpthread
[root@localhost google]# ./san
[root@localhost google]#

View All Answers

Question - 79:

What is the output of this program?
```
    #include<stdio.h>
    #include<pthread.h>

    void *fun_t(void *arg);
    void *fun_t(void *arg)
    {
        int ret;
        ret = pthread_exit("Bye");
        printf("%dn",ret);
    }
    int main()
    {
        pthread_t pt;
        void *res_t;
        if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
```

```
     perror("pthread_create");
  if(pthread_join(pt,&res_t) != 0)
     perror("pthread_join");
  return 0;
}
```
a) 0
b) 1
c) -1
d) none of the mentioned

Ans:

d) none of the mentioned
Explanation:
The function pthread_exit() does not return any value. Hence this program will give an error.
Output:
[root@localhost Google]# gcc -o san san.c -lpthread
san.c: In function 'fun_t':
san.c:8:6: error: void value not ignored as it ought to be
[root@localhost google]#

<u>View All Answers</u>

Question - 80:

Which one of the following string will print first by this program?
```
  #include<stdio.h>
  #include<pthread.h>

  void *fun_t(void *arg);
  void *fun_t(void *arg)
  {
    printf("Googlen");
    pthread_exit("Bye");
  }
  int main()
  {
    pthread_t pt;
    void *res_t;
    if(pthread_create(&pt,NULL,fun_t,NULL) != 0)
      perror("pthread_create");
    printf("Linuxn");
    if(pthread_join(pt,&res_t) != 0)
      perror("pthread_join");
    return 0;
  }
```
a) Linux
b) Google
c) it can not be predicted
d) none of the mentioned

Ans:

b) Google
Explanation:It depends upon the scheduler.
Output:
[root@localhost Google]# gcc -o san san.c -lpthread
[root@localhost Google]# ./san
Google
Linux
[root@localhost threads]#

<u>View All Answers</u>

Linux OS Most Popular & Related Interview Guides

1 : **Device Drivers Interview Questions and Answers.**

2 : **Linux OS Management Interview Questions and Answers.**

3 : **Linux Makefile Interview Questions and Answers.**

4 : **Linux Environment Interview Questions and Answers.**

5 : **Linux OS Shell Interview Questions and Answers.**

6 : **GCC Compiler Interview Questions and Answers.**

7 : **Linux OS Interview Questions and Answers.**

8 : **Bash Arithmetic Expressions Interview Questions and Answers.**

9 : **Linux IPC Interview Questions and Answers.**

10 : **System Calls Interview Questions and Answers.**