

**Question 1**

Partially correct

Mark 0.25 out of  
1.00[Flag question](#)

Select all statements that correctly explain the use/purpose of system calls.

Select one or more:

- a. Allow I/O device access to user processes
- b. Handle exceptions like division by zero
- c. Switch from user mode to kernel mode✓
- d. Run each instruction of an application program
- e. Handle ALL types of interrupts
- f. Provide services for accessing files
- g. Provide an environment for process creation

Your answer is partially correct.

You have correctly selected 1.

The correct answers are: Switch from user mode to kernel mode, Provide services for accessing files, Allow I/O device access to user processes, Provide an environment for process creation

**Question 2**

Not answered

Marked out of  
1.00[Flag question](#)

Select the sequence of events that are NOT possible, assuming a non-interruptible kernel code

(Note: non-interruptible kernel code means, if the kernel code is executing, then interrupts will be disabled).

Note: A possible sequence may have some missing steps in between. An impossible sequence will have n and n+1th steps such that n+1th step can not follow n'th step.

Select one or more:

- a.
  - P1 running
  - P1 makes system call
  - Scheduler
  - P2 running
  - P2 makes system call and blocks
  - Scheduler
  - P1 running again
- b. P1 running
  - P1 makes system call and blocks
  - Scheduler
  - P2 running
  - P2 makes system call and blocks
  - Scheduler
  - P1 running again
- c. P1 running
  - P1 makes system call
  - system call returns
  - P1 running

Your answer is incorrect.

The correct answers are: P1 running  
P1 makes system call and blocks  
Scheduler  
P2 running  
P2 makes system call and blocks  
Scheduler  
P1 running again, P1 running  
P1 makes system call  
timer interrupt  
Scheduler  
P2 running  
timer interrupt  
Scheduler  
P1 running  
P1's system call return,  
P1 running  
P1 makes system call  
Scheduler  
P2 running  
P2 makes system call and blocks  
Scheduler  
P1 running again

**Question 3**

Correct

Mark 0.50 out of  
0.50

[Flag question](#)

Compare multiprogramming with multitasking

- a. A multitasking system is not necessarily multiprogramming
- b. A multiprogramming system is not necessarily multitasking✓

The correct answer is: A multiprogramming system is not necessarily multitasking

**Question 4**

Not answered

Marked out of  
1.00

[Flag question](#)

How does the distinction between kernel mode and user mode function as a rudimentary form of protection (security) ?

Select one:

- a. It prohibits a user mode process from running privileged instructions
- b. It prohibits one process from accessing other process's memory
- c. It prohibits invocation of kernel code completely, if a user program is running
- d. It disallows hardware interrupts when a process is running

Your answer is incorrect.

The correct answer is: It prohibits a user mode process from running privileged instructions

**Question 5**

Correct

Mark 0.50 out of  
0.50[Flag question](#)

When you turn your computer ON, on BIOS based systems, you are often shown an option like "Press F9 for boot options". What does this mean?

- a. The choice of which OS to boot from
- b. The choice of booting slowly or fast
- c. The choice of the boot loader (e.g. GRUB or Windows-Loader)
- d. The BIOS allows us to choose the boot device, the device from which the boot loader will be loaded ✓

The correct answer is: The BIOS allows us to choose the boot device, the device from which the boot loader will be loaded

**Question 6**

Correct

Mark 1.00 out of  
1.00[Flag question](#)

Predict the output of the program given here.

Assume that all the path names for the programs are correct. For example "/usr/bin/echo" will actually run echo command.

Assume that there is no mixing of printf output on screen if two of them run concurrently.

In the answer replace a new line by a single space.

For example::

good

output

~~should be written as good output~~

~~and the answer replace a new line by a single space.~~

For example::

good

output

~~should be written as good output~~

--

```
main() {  
    int i;  
    i = fork();  
    if(i == 0)  
        execl("/usr/bin/echo", "/usr/bin/echo", "hi", 0);  
    else  
        wait(0);  
    fork();  
    execl("/usr/bin/echo", "/usr/bin/echo", "one", 0);  
}
```

Answer:



**Question 7**

Correct

Mark 0.50 out of  
0.50[Flag question](#)

Is the terminal a part of the kernel on GNU/Linux systems?

- a. no  wrong
- b. yes

The correct answer is: no

**Question 8**

Incorrect

Mark 0.00 out of  
1.00[Flag question](#)

Write the possible contents of the file /tmp/xyz after this program.

In the answer if you want to mention any non-text character, then write \0. For example abc\0\0 means abc followed by any two non-text characters

```
int main(int argc, char *argv[]) {  
    int fd1, fd2, n, i;  
    char buf[128];  
  
    fd1 = open("/tmp/xyz", O_WRONLY | O_CREAT, S_IRUSR|S_IWUSR);  
    write(fd1, "hello", 5);  
    fd2 = open("/tmp/xyz", O_WRONLY, S_IRUSR|S_IWUSR);  
    write(fd2, "bye", 3);  
    close(fd1);  
    close(fd2);  
    return 0;  
}
```

Answer:  

The correct answer is: byelo

**Question 9**

Partially correct

Mark 0.75 out of  
1.00[Flag question](#)

Select all the correct statements about the process init on Linuxes/Unixes.

Select one or more:

- a. init is created by kernel by forking itself
- b. any user process can fork and exec init
- c. no process can exec 'init'
- d. init can not be killed with SIGKILL
- e. only a process run by 'root' user can exec 'init' ✓
- f. init typically has a pid=1 ✓
- g. init is created by kernel 'by hand' ✓

Your answer is partially correct.

You have correctly selected 3.

The correct answers are: init is created by kernel 'by hand', init typically has a pid=1, init can not be killed with SIGKILL, only a process run by 'root' user can exec 'init'

**Question 10**

Correct

Mark 1.00 out of  
1.00[Flag question](#)

Order the following events in boot process (from 1 onwards)

Login interface	5	▼	✓
Shell	6	▼	✓
Boot loader	2	▼	✓
Init	4	▼	✓
OS	3	▼	✓
BIOS	1	▼	✓

Your answer is correct.

The correct answer is: Login interface → 5, Shell → 6, Boot loader → 2, Init → 4, OS → 3, BIOS → 1

**Question 11**

Correct

Mark 1.00 out of  
1.00[Flag question](#)**What will this program do?**

```
int main() {
    fork();
    execl("/bin/ls", "/bin/ls", NULL);
    printf("hello");
}
```

- a. run ls once
- b. run ls twice and print hello twice, but output will appear in some random order
- c. run ls twice✓
- d. run ls twice and print hello twice
- e. one process will run ls, another will print hello

Your answer is correct.

The correct answer is: run ls twice

**Question 12**

Partially correct

Mark 0.75 out of  
1.00[Flag question](#)**Given below is the output of "ps -ef".****Answer the questions based on it.**

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	Jan05	?	00:01:08	/sbin/init splash
root	2	0	0	Jan05	?	00:00:00	[kthreadd]
root	3	2	0	Jan05	?	00:00:00	[rcu_gp]
root	4	2	0	Jan05	?	00:00:00	[rcu_par_gp]
root	9	2	0	Jan05	?	00:00:00	[mm_percpu_wq]
root	10	2	0	Jan05	?	00:00:00	[rcu_tasks_rude_]
root	11	2	0	Jan05	?	00:00:00	[rcu_tasks_trace]
root	12	2	0	Jan05	?	00:00:22	[ksoftirqd/0]
root	13	2	0	Jan05	?	00:06:29	[rcu_sched]
root	14	2	0	Jan05	?	00:00:02	[migration/0]
root	15	2	0	Jan05	?	00:00:00	[idle_inject/0]
root	16	2	0	Jan05	?	00:00:00	[cpuhp/0]
root	17	2	0	Jan05	?	00:00:00	[cpuhp/1]
root	18	2	0	Jan05	?	00:00:00	[idle_inject/1]
root	19	2	0	Jan05	?	00:00:03	[migration/1]
root	20	2	0	Jan05	?	00:00:13	[ksoftirqd/1]
root	22	2	0	Jan05	?	00:00:00	[kworker/1:0H-events_highpri]
root	23	2	0	Jan05	?	00:00:00	[cpuhp/2]
root	24	2	0	Jan05	?	00:00:00	[idle_inject/2]
root	25	2	0	Jan05	?	00:00:01	[migration/2]

```

abhijit 977184 594726 0 15:21 pts/5 00:00:00 ps -eaf
abhijit 1664880 3629 0 Jan07 ? 00:00:06 /usr/bin/gnome-calendar --
gapplication-service
abhijit 1665340 3629 0 Jan07 ? 00:00:00 /usr/bin/gpg-agent --supervised
abhijit 3872409 3629 0 Jan07 ? 00:00:05 /usr/bin/seahorse --gapplication-
service
root 3873244 1 0 Jan07 ? 00:00:55 /sbin/mount.ntfs /dev/nvme0n1p3 /
media/abhijit/windows -o rw,nodev,nosuid,windows_names,uid=1000,gid=1000,uhelper=udisks2
abhijit 3884359 30565 0 Jan07 pts/7 00:00:00 bash
abhijit 4108623 30565 0 Jan08 pts/8 00:00:00 bash
abhijit 4136834 3890 0 Jan08 ? 00:00:00 /usr/lib/libreoffice/program/oosplash
--calc
abhijit 4136869 4136834 0 Jan08 ? 00:32:11 /usr/lib/libreoffice/program/
soffice.bin --calc
abhijit 4139495 30565 0 Jan08 pts/9 00:00:00 bash

```

The PID of the grand-parent of the process with PID 4108623 is : 3629 ✓

The two processes which were created by kernel have PIDs (in increasing order) 1 ✓ and 2 ✓

The process that created most of the "graphical" processes is having PID 2 ✗

### Question 13

Correct

Mark 0.50 out of  
0.50

 Flag question

Select all the correct statements about bootloader.

Every wrong selection will deduct marks proportional to  $1/n$  where  $n$  is total wrong choices in the question.

You will get minimum a zero.

- a. Bootloader must be one sector in length
- b. LILO is a bootloader ✓
- c. The bootloader loads the BIOS
- d. Bootloaders allow selection of OS to boot from ✓
- e. Modern Bootloaders often allow configuring the way an OS boots ✓

Your answer is correct.

The correct answers are: LILO is a bootloader, Modern Bootloaders often allow configuring the way an OS boots,  
Bootloaders allow selection of OS to boot from

**Question 14**

Partially correct

Mark 0.60 out of  
1.00[Flag question](#)

Select all the correct statements about two modes of CPU operation

Select one or more:

- a. Some instructions are allowed to run only in user mode, while all instructions can run in kernel mode ✓
- b. There is an instruction like 'iret' to return from kernel mode to user mode ✓
- c. The software interrupt instructions change the mode from user mode to kernel mode and jumps to predefined ✓ location simultaneously
- d. The two modes are essential for a multiprogramming system
- e. The two modes are essential for a multitasking system

Your answer is partially correct.

You have correctly selected 3.

The correct answers are: The two modes are essential for a multiprogramming system, The two modes are essential for a multitasking system, There is an instruction like 'iret' to return from kernel mode to user mode, The software interrupt instructions change the mode from user mode to kernel mode and jumps to predefined location simultaneously, Some instructions are allowed to run only in user mode, while all instructions can run in kernel mode

**Question 15**

Correct

Mark 1.00 out of  
1.00[Flag question](#)

Consider the following programs

**exec1.c**

```
#include <unistd.h>
#include <stdio.h>
int main() {
    exec("./exec2", "./exec2", NULL);
}
```

**exec2.c**

```
#include <unistd.h>
#include <stdio.h>
int main() {
    exec("/bin/ls", "/bin/ls", NULL);
    printf("hello\n");
}
```

Compiled as

```
cc  exec1.c -o exec1
cc  exec2.c -o exec2
```

And run as

```
$ ./exec1
```

**Explain the output of the above command (./exec1)****Assume that /bin/ls , i.e. the 'ls' program exists.**

---

**Select one:**

```
cc exec1.c -o exec1  
cc exec2.c -o exec2
```

And run as

```
$ ./exec1
```

Explain the output of the above command (./exec1)

Assume that /bin/ls , i.e. the 'ls' program exists.

Select one:

- a. "ls" runs on current directory ✓
- b. Execution fails as the call to execl() in exec2 fails
- c. Program prints hello
- d. Execution fails as the call to execl() in exec1 fails
- e. Execution fails as one exec can't invoke another exec

Your answer is correct.

The correct answer is: "ls" runs on current directory

**Question 16**

Partially correct

Select all the correct statements about two modes of CPU operation

**Question 16**

Partially correct

Mark 0.60 out of  
1.00

 [Flag question](#)

Select all the correct statements about two modes of CPU operation

Select one or more:

- a. Some instructions are allowed to run only in user mode, while all instructions can run in kernel mode ✓
- b. There is an instruction like 'iret' to return from kernel mode to user mode ✓
- c. The software interrupt instructions change the mode from user mode to kernel mode and jumps to predefined ✓ location simultaneously
- d. The two modes are essential for a multitasking system
- e. The two modes are essential for a multiprogramming system

Your answer is partially correct.

You have correctly selected 3.

The correct answers are: The two modes are essential for a multiprogramming system, The two modes are essential for a multitasking system, There is an instruction like 'iret' to return from kernel mode to user mode, The software interrupt instructions change the mode from user mode to kernel mode and jumps to predefined location simultaneously, Some instructions are allowed to run only in user mode, while all instructions can run in kernel mode

**Question 17**

Correct

Mark 1.00 out of  
1.00

 Flag question

Select all the correct statements about bootloader.

Every wrong selection will deduct marks proportional to  $1/n$  where n is total wrong choices in the question.

You will get minimum a zero.

- a. Bootloader must be one sector in length
- b. LILO is a bootloader✓
- c. Bootloaders allow selection of OS to boot from✓
- d. Modern Bootloaders often allow configuring the way an OS boots✓
- e. The bootloader loads the BIOS

Your answer is correct.

The correct answers are: LILO is a bootloader, Modern Bootloaders often allow configuring the way an OS boots, Bootloaders allow selection of OS to boot from





