

# Unit 2 - Week 1

## Course outline

How does an NPTEL online course work?

### Week 1

- Introductio to UNIX System Calls Part - 1
- Introductio to UNIX System Calls Part - 2
- Threads, Address Spaces, Filesystem Devices

### Quiz : Assignment 1

### Week 1 Feedback Form

### Week 2

### Week 3

### Week 4

### Week 5

### Week 6

### Week 7

### Week 8

### Week 9

### Week 10

### Week 11

### Week 12

### Assignment Solution

### Download Videos

### Text Transcripts

# Assignment 1

The due date for submitting this assignment has passed.  
As per our records you have not submitted this assignment.

**Due on 2020-02-12, 23:59 IST.**

1) Which of the following is typically a part of the operating system but not the kernel? 1 point

- Graphical User Interface
- Network Management
- Device Driver Management
- Compiler
- Utilities such as ls, chmod and chown

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Graphical User Interface  
Compiler  
Utilities such as ls, chmod and chown

2) The "seek" system call allows the application program to change the value of the file's offset so that subsequent read/write is performed from a new position in the file. Which of the following task will require the use of seek operation? 1 point

- Copying the contents of file A to B
- Reversing the contents of a file
- Insert/update/delete at a particular point
- Finding a particular character in a file

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Reversing the contents of a file  
Insert/update/delete at a particular point

3) Which of the following can have an operating system? 1 point

- Microprocessor
- Car
- Phone
- Microcontroller
- Watches

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Microprocessor  
Car  
Phone  
Watches

4) Which of the following is true about shell? 1 point

- Term "terminal" is synonymous to shell
- Bash is synonymous to shell
- Shells are ought to be part of the operating system
- Users can install third party shells to replace ones shipped with OS if any

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Users can install third party shells to replace ones shipped with OS if any

5) Which of the following is abstracted by operating system? 1 point

- Processor
- Memory
- Network Cards
- All of the above

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
All of the above

6) Which of the following are valid differences between CreateProcess() and fork(): 1 point

- fork() by default creates a child process with same file descriptors while CreateProcess() does not.
- CreateProcess() by default creates a child process with same file descriptors while fork() does not.
- fork() duplicates the program for different process. CreateProcess() creates different process with new program.
- CreateProcess() does not return the pid of the child process to the parent process while fork() returns the child process pid to parent process.
- CreateProcess() is more efficient than fork() then exec() without copy-on-write.

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
fork() by default creates a child process with same file descriptors while CreateProcess() does not.  
fork() duplicates the program for different process. CreateProcess() creates different process with new program.  
CreateProcess() is more efficient than fork() then exec() without copy-on-write.

7) An operating system with multiprogramming capability is one that 1 point

- allows several users to use the same program at once by giving each a slice of time
- loads several independent processes into memory and switches the CPU from one job to another as required
- runs programs over more than one processor
- None of the above

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
loads several independent processes into memory and switches the CPU from one job to another as required

8) How does the shell implement "&", backgrounding? e.g., \$ ./compute & 1 point

- No change required in the shell implementation as shown in the lecture video
- Using sleep() syscall for defined time.
- Not calling wait syscall
- Not calling wait syscall followed by initiating SIGCHLD handler, which gets invoked after termination of child process
- Not calling wait syscall followed by initiating SIGCHLD handler, which gets invoked at the start of child process
- Cannot be implemented without making changes in the process scheduling mechanism of OS

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Not calling wait syscall followed by initiating SIGCHLD handler, which gets invoked after termination of child process

9) What are the standard file descriptor numbers for STDERR, STDIN, and STDOUT? 1 point

- 1,2,3
- 0,1,2
- 2,0,1
- Randomly decide

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
2,0,1

10) Consider the following code: 1 point

```
for (i = 0; i < 4; i++)
{
    fork();
}
```

If we start with one process, what is the total number of processes spawned by this loop (excluding the first process).

- 4
- 15
- 16
- 14

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
15

11) Consider two implementations of 2 >& 1 (i.e redirecting ERR to OUTPUT file location): 1 point

```
1 // Implementation A:
2
3 close(1);
4 open("output_file_A");
5 close(2);
6 open("output_file_A");
7 write(1, "operating",9);
8 write(2, "system",6);
9
10 // Implementation B:
11
12 close(1);
13 open("output_file_B");
14 close(2);
15 dup(1);
16 write(1, "operating",9);
17 write(2, "system",6);
```

Which of the following options are correct for above implementations?

- Output\_file\_A content: "operatingsystem" and Output\_file\_A content: "system"
- The offset for output\_file\_A is 6 and offset for output\_file\_B is 15
- Output\_file\_A content: "system" and Output\_file\_A content: "operatingsystem"
- The offset for output\_file\_A is 15 and offset for output\_file\_B is 6

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
The offset for output\_file\_A is 6 and offset for output\_file\_B is 15  
Output\_file\_A content: "system" and Output\_file\_A content: "operatingsystem"

12) How many times the following C program prints yes? 1 point

```
main()
{
    fork();
    fork();
    printf("yes");
}
```

- Only once
- Twice
- Four times
- Eight times

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
Four times

13) consider the following program: 1 point

```
//Program A:
main()
{
    int fd;
    fork();
    fd = open("outfile_A", O_RDWR);
    write(fd, "hello", 5);
    exit();
}
```

```
//Program B:
main()
{
    int fd;
    fork();
    fd = open("outfile_B", O_RDWR);
    write(fd, "hello", 5);
    exit();
}
```

Assume all system calls finish successfully on a uniprocessor system. Also, assume that a system call cannot be interrupted in the middle of its execution. What will be the contents of the "outfile\_A" and "outfile\_B" file, after all processes have successfully exited?

- "hellohello" and "hellohello"
- "hellohello" and "hello"
- "hello" and "hellohello"

No, the answer is incorrect.  
Score: 0

Accepted Answers:  
"hello" and "hellohello"