Branch : CSE & IT **Batch : English**

Operating Systems

Miscellaneous

DPP 01

[NAT]

1. Consider the below code segment. Total new process created is _______

```
#include<stdio.h>

int main ()

{

   int i;

   for (i=0; i<5, i+=2)

{

   switch(i)

   {

        Case 0: fork();

        Case 1: fork(); fork();

        Case 2: fork(); fork(); break;

        Case 3: fork(); fork(); fork(); break;

        Case 4: for (j = i, j >= 1, j --) fork();

   }

 }

return 0;

}
```

[NAT]

2. Consider the following code:

```
main()

{

   for(int i = 1, i<= 5; i++)

   fork ()

}
```

What will be the number of child processes/newly created processes for the above code?

[MCQ]

3. Match List – I and List – II and select the correct answer using the code given below:

List – 1	List – II
A. Context switching	1. Process creation
B. Degree of multi programming	2. Dispatcher
C. Message passing	3. Long term scheduler
D. Fork()	4. Inter process communication

Codes:

	A	B	C	D
(a)	3	1	2	4
(b)	2	3	4	1
(c)	1	3	4	2
(d)	4	2	1	3

[MCQ]

4. To access the services of operating system, the interface is provided by the

(a) System call

(b) API

(c) Library

(d) Assembly Instructions.

[MCQ]

5. The following program fragment prints the strings “GATE2023” how many number of times?

```
  main ()
    {
          fork ();
          fork ();
          printf(“GATE2023”\n);
    }
```

(a) 1 (b) 2

(c) 4 (d) 8

[MCQ]

6. Which of the following scheduling can be done by thread library?

(a) User thread scheduling

(b) Process Scheduling

(c) Kernel thread scheduling

(d) None of the above.

[MCQ]

7. Consider the following code:

```
void main()
        {
            for (int k = 1; k < 5; k++)
            pid[k] = fork();
        }
```

How many child processes created by the above code:

(a) 15 (b) 4

(c) 32 (d) 16

[MCQ]

8. Consider the following statements with respect to user-level threads and kernel – level threads.

S_1: Context switching in kernel level threads is faster than user level threads.

S_2: If one thread of user – level gets blocked, entire process also gets blocked.

S_3: kernel – level threads can be scheduled independently.

(a) S_1 and S_3 only

(b) Only S_2

(c) Only S_3

(d) S_2 and S_3 only

[MSQ]

9. What are the types of system calls in an operating system?

(a) Process control, file management

(b) Device management.

(c) Communication.

(d) Information maintenance.

[MCQ]

10. Which of the following operations require the executing code to be operating with Kernal mode?

(a) Performing semaphore ‘P’ operation

(b) Making system call

(c) Disabling interrupt

(d) Both (a) and (c)

Answer Key

1. **(2047)**
2. **(31)**
3. **(b)**
4. **(a)**
5. **(c)**
6. **(a)**
7. **(a)**
8. **(d)**
9. **(a, b, c, d)**
10. **(c)**

Hints & Solutions

1. (2047)

i = 0; 0 < 5 true switch(0)

Case 0: fork();

Case 1: fork(); fork();

Case 2: fork(); fork(); break; i = 2

i = 2; 2 < 5 true switch(2)

Case 2: fork(); fork(); break; i = 4

i = 4; 4 < 5 true switch(4)

Case 4: for j = 4 to 1 fork (); fork (); fork (); fork (); i = 6.

i = 5; 6 < 5 false

Total fork() calls made = 11

Total new processes = $2^{11} – 1 = 2047$

2. (31)

for n fork calls, number of child processes created = $2^n – 1$

$\therefore 2^5 – 1$

$= 32 – 1$

$= 31.$

3. (b)

- Context switching: Whenever process dispatch to running state by dispatcher, we need to perform context switching (loading PCB)
- Degree of multiprogramming: long term scheduler is responsible for creating new process to main memory so, it controls degree of multiprogramming.
- Message passing: Inter process communication (IPC) uses message passing method to communicate each other.
- Fork() : System call used to create new child process.

4. (a)

Through system call a process requests a service from the kernel of the operating system on which it is executing:

```
main ()
    {
        printf(“Hello world”);
    }
```

Here “printf()” is a system call used to request for monitor for printing “Hello world”.

5. (c)

- If fork() calls “n” number of times then “2^n” number of process created.

  So, Total process created = $2^2 = 4$ processes
- Above program print 4 times

  “GATE2023”.

6. (a)

Programmer can create and manage threads using API provided by thread library.

7. (a)

- Fork() is a system call used to create the child process.
- If fork() called “n” time then total number of child process will be “$2^n – 1$”.
- Total “2^n”; 1 process is parent process and “2^n–1” are child processes.
- Above program can be re-written as

```
void main()
{
    fork ();
    fork ();
    fork ();
    fork ();
}
```

- fork () called “4” times so, total child process will be “2^4–1” = 15.

8. (d)

S_1: Kernel – level threads have more context than user-level so, context switching in kernel – level threads is slower than user – level.

S_2: User – level threads are created using software so, operating system can not differentiate between user – level threads. So, blocking one user-level thread block entire process. True.

S_3: Operating system can easily differentiate kernel – level thread so, kernel – level threads can be scheduled independently. True.

9. (a, b, c, d)

There are five types of system calls – process control, file management

Device management, information management, communication.

10. (c)

System calls are executed in kernel mode but not made in kernel mode i.e. made in user mode.

Disabling interrupt is done in kernel mode.

So, option (c) is answer.