Mestrado em Engenharia de Sistemas Computacionais Críticos
Programação de Sistemas Operativos Tempo-Real (RTOSP)
2022/2023

Normal Period	**Regular Exam**	**Duration**	**Version**
03/07/2023	With no consultation	1h:00m	A

Number:_____ **Name**:_____

1. Concerning the Operating System (OS):
   (a) The computer programs make use of the OS by making requests for services through a defined application programming interface (API);
   (b) OS(es) abstract the physical resources;
   (c) All of the above;
   (d) None of the above.

2. In a multiprocessing system:
   (a) Parallel computing is achievable;
   (b) Parallel computing is not achievable;
   (c) All of the above;
   (d) None of the above.

3. General-Purpose Operating System (GPOS)
   (a) It use a fairness policy to carry processes and threads to the Central Processing Unit (CPU);
   (b) It is designed to perform non-time-critical tasks;
   (c) All of the above;
   (d) None of the above.

4. The instruction cycle is the time required by the CPU to execute one single instruction and it is composed by:
   (a) Decode, Execute and Store;
   (b) Fetch, Execute and Store;
   (c) All of the above;
   (d) None of the above.

5. The Linux Kernel:
   (a) It is preemptive;
   (b) It is monolithic;
   (c) All of the above;
   (d) None of the above.
6. An interrupt:
   (a) It is an event that alters the sequence of instructions executed by a CPU;
   (b) Can be produced by a CPU;
   (c) All of the above;
   (d) None of the above.
7. Loadable kernel modules:
   (a) Can be inserted into kernel and removed from kernel without rebooting;
   (b) Require Linux kernel compilation:
   (c) All of the above;
   (d) None of the above.
8. The Linux kernel build system:
   (a) Uses configuration symbols to conditionally compile the Linux kernel;
   (b) It builds the object code by recursively descending into the subdirectories of the kernel source tree;
   (c) All of the above;
   (d) None of the above.
9. From the schedule point of view, the Linux kernel:
   (a) Does not differentiate threads and processes;
   (b) Differentiates threads and processes;
   (c) All of the above;
   (d) None of the above.
10. In the Linux kernel:
   (a) A System call is identified by a number;
   (b) A System call is identified by an interrupt;
   (c) All of the above;
   (d) None of the above.

11. Process/thread identification:
    (a) When a new process is created (using `fork`), the `PID` and `TGID` identifiers have the same value;
    (b) When a new thread is created (using `create_thread`), it inherits the `TGID` identifier from creator and gets its own `PID` identifier;
    (c) All of the above;
    (d) None of the above.
12. In Linux kernel, the process/thread scheduling:
    (a) Is organized by priorities;
    (b) Uses scheduling classes to implement scheduling policies;
    (c) All of the above;
    (d) None of the above.
13. In Linux kernel, each process/thread at a given time instant:
    (a) Is assigned to only one scheduling class;
    (b) Is assigned to an arbitrary number of scheduling classes;
    (c) All of the above;
    (d) None of the above.
14. The Linux kernel:
    (a) Creates a `struct rq` instance per CPU;
    (b) Creates one `struct rq` instance for all CPUs;
    (c) All of the above;
    (d) None of the above.
15. The Linux kernel provides an implementation of red-black tree, called `rbtree`:
    (a) It is suitable for implementing priority based mechanisms;
    (b) It is used by Real-Time (RT) scheduling class;
    (c) All of the above;
    (d) None of the above.
16. In the Linux kernel, there is an idle task:
    (a) For the system;
    (b) Scheduled according to the `SCHED_IDLE` scheduling policy;
    (c) All of the above;
    (d) None of the above.

17. In the Linux kernel, whenever a process is in `TASK_RUNNING` state:

    (a) It is being executed (i.e., it is assigned to the CPU for execution);

    (b) It is ready for execution in a waitqueue;

    (c) All of the above;

    (d) None of the above.

18. In the Linux kernel, the macro `container_of (ptr, type, member)`:

    (a) Retrieves the value of the respective member;

    (b) Retrieves the address of the container which accommodates the respective member;

    (c) All of the above;

    (d) None of the above.

19. The Linux kernel tick rate is defined by `HZ`:

    (a) A tick is a timer interrupt;

    (b) At every tick the scheduler core function (`__schedule`) is invoked;

    (c) All of the above;

    (d) None of the above.

20. In the Linux kernel, synchronization mechanisms:

    (a) Semaphore's value can be updated by any thread/process;

    (b) Only the thread mutex owner can release it;

    (c) All of the above;

    (d) None of the above.