

Course description:

The lecture will focus on getting to know the functions of the kernel of the operating system available to the system programmer. In addition to themes common to most SO we will get acquainted with the solutions available in Unix and Unix-like systems (Linux, FreeBSD, NetBSD, MacOS, QNX, MINIX3).

Introduction [4h]

- The kernel, the operating system distribution.
- Types of operating systems (built-in, real-time, distributed).
- The most important achievements in the field of operating systems.
- System calls and communication programs to the kernel.
- Operating system kernel architecture (monolithic kernel, microkernel, exokernel).

Processes and threads [8h]

- The process address space.
- Management processes and threads.
- The hierarchy of processes.
- Scheduling tasks.
- Synchronization (disable interrupts, locks, condition variables).
- Communications (pipes, sockets, mailboxes)
- Concurrent programming (temporal logic).

Memory [8h]

- Management of physical memory (twin system).
- Letter allocation algorithm.
- Pools objects.
- Virtual memory.
- Managing the virtual space.
- Paging on demand caching.
- Page replacement algorithms.
- Loading programs, shared libraries.
- Versioning address space (vm_object).

Files, file systems [6h]

- Operations on files and directories.
- The hierarchy of files, mounting points.
- File permissions.
- Virtual File System (VFS).
- Managing disk space.
- Organization of file systems (i-nodes, directories).

- Paddles, logs, backups.

Additional topics [4h]

- Drivers - resources, equipment, rails, device discovery process.
- System clock and time management.
- The process of loading the operating system.
- Authorization and authentication (PAM).
- Preventing cyber attacks.
- Virtual machines.