

### 3. Режим користувача і режим ядра



# 3. Режим користувача і режим ядра

**Режим ядра** (*kernel mode*; захищений режим, привілейований режим)

Процес може використати будь-яку інструкцію, яку здатний виконати даний комп'ютер.

У режимі ядра працює операційна система.

**Режим користувача** (*user mode*)

Процес має доступ до обмеженої підмножини інструкцій даного комп'ютера.

У режимі користувача працює решта програм.

**Ядром ОС** називається сукупність базових компонентів ОС, які

виконують її найважливіші функції,

працюють у привілейованому режимі

і зазвичай постійно перебувають у пам'яті.

Коли програмі потрібно виконати дію, реалізовану у ядрі, вона виконує **системний виклик** (system call).

# 3. Режим користувача і режим ядра

## *Базові тези про системні виклики*

- Найчастіше системні виклики написані мовами C, C++, Assembler.
- Системні виклики скрізь:
  - для роботи з файлами (відкриття, закриття, читання, запис, створення, вилучення, читання/запис атрибутів тощо);
  - для роботи з пристроями (приклади: вивести щось на екран, зчитати введену користувачем команду, ...);
  - для роботи з процесами (створення, завершення, виділення/вивільнення пам'яті, читання/запис атрибутів тощо);
  - для обміну повідомленнями всередині системи і т. д.
- Логіка роботи системного виклику - як у звичайної підпрограми (функції):
  - передавання параметрів і керування - у підпрограму;
  - виконання підпрограми;
  - повернення керування і результатів у точку виклику.
- На відміну від звичайної підпрограми, відбувається перемикання між режимами користувача і ядра. Це потребує більше часу.

# Для самоїтнього читання

1. [*Silberschatz, Galvin, Gagne, 2018*] Chapter 3 (paragr. 3.1).
2. [*Stollings, 2017*] Chapters 3-4.
3. [*Tanenbaum, Bos, 2014*] Chapter 2 (paragr. 2.1) + Chapter 1 (paragr. 1.4)
4. [*Шеховцов, 2009*] Розділи 2-3.