

## Лекція 1

# АРХІТЕКТУРА, ПРИЗНАЧЕННЯ І ФУНКЦІЇ ОПЕРАЦІЙНИХ СИСТЕМ

### 1.1 Функції, класифікації і складові ОС

*Операційна система (ОС)* – це програма, що керує роботою інших програм, як прикладних, так і системних, і є інтерфейсом між програмами і апаратним забезпеченням комп'ютера. Тому, можемо стверджувати, що найголовнішими функціями, які виконує операційна система є:

*Інтерфейс між користувачем і комп'ютером.*

*Керування ресурсами.*

Також виділяють і інші функції операційних систем.

#### ФУНКЦІЇ ОС

##### ОСНОВНІ

Виконання на вимогу програм користувача тих елементарних (низькорівневих) дій, які є спільними для більшості програмного забезпечення і часто зустрічаються майже у всіх програмах (введення та виведення даних, запуск і зупинка інших програм, виділення та вивільнення додаткової пам'яті тощо).  
Стандартизований доступ до периферійних пристроїв (пристроїв вводу-виводу).  
Завантаження програм в оперативну пам'ять і їх виконання.  
Керування оперативною пам'яттю (розподіл між процесами, організація віртуальної пам'яті).  
Керування доступом до даних енергонезалежних носіїв (жорсткий диск, оптичні диски, тощо), організованим у тій чи іншій файлової системі.  
Забезпечення користувацького інтерфейсу.

##### ДОДАТКОВІ

Паралельне або псевдопаралельне виконання задач (багатозадачність).  
Розподіл ресурсів обчислювальної системи між процесами.  
Організація надійних обчислень, що полягає у розмежуванні доступу до ресурсів. Тобто унеможливлення впливу процесу на перебіг інших процесів та на сам ресурс.  
Взаємодія між процесами: обмін даними, синхронізація.  
Захист самої системи, а також користувацьких даних і програм від дій користувача або програм.  
Багатокористувацький режим роботи та розділення прав доступу (автентифікація, авторизація).

<sup>1</sup> При підготовці даного розділу були використані літературні джерела [1–4], [9–11], [12], [15], [17], [21], [23] та ін.

Пересічним користувачем комп'ютерна система сприймається, як набір додатків (прикладних програм) і його не цікавлять подробиці апаратного забезпечення комп'ютерної системи. Додатки створюють програмісти на одній мов програмування і для спрощення програмування функцій управління комп'ютером розроблено цілий набір системних програм (деякі з них називаються утилітами), які реалізують часто використовувані функції (роботу файлами, управління пристроями введення/виведення і т. д.). Програмісти використовують ці засоби при розробці власних програм. Зрозуміло, що найбільш важливою частиною системних програм є ОС, що безпосередньо взаємодіє з апаратурою. Отже, маємо сприймати ОС як посередника, що забезпечує і спрощує користувачеві (програмісту), а також програмним додаткам доступ до різних служб, зменшує залежність від конкретної реалізації апаратного забезпечення. Також ОС управляє ресурсами, розподіляючи їх між програмами, або у часі (розподіл доступу до процесора, принтеру), або в просторі (використання оперативної пам'яті, дискового простору).

На рисунку 1.1 подана ієрархічна структура програмного забезпечення.

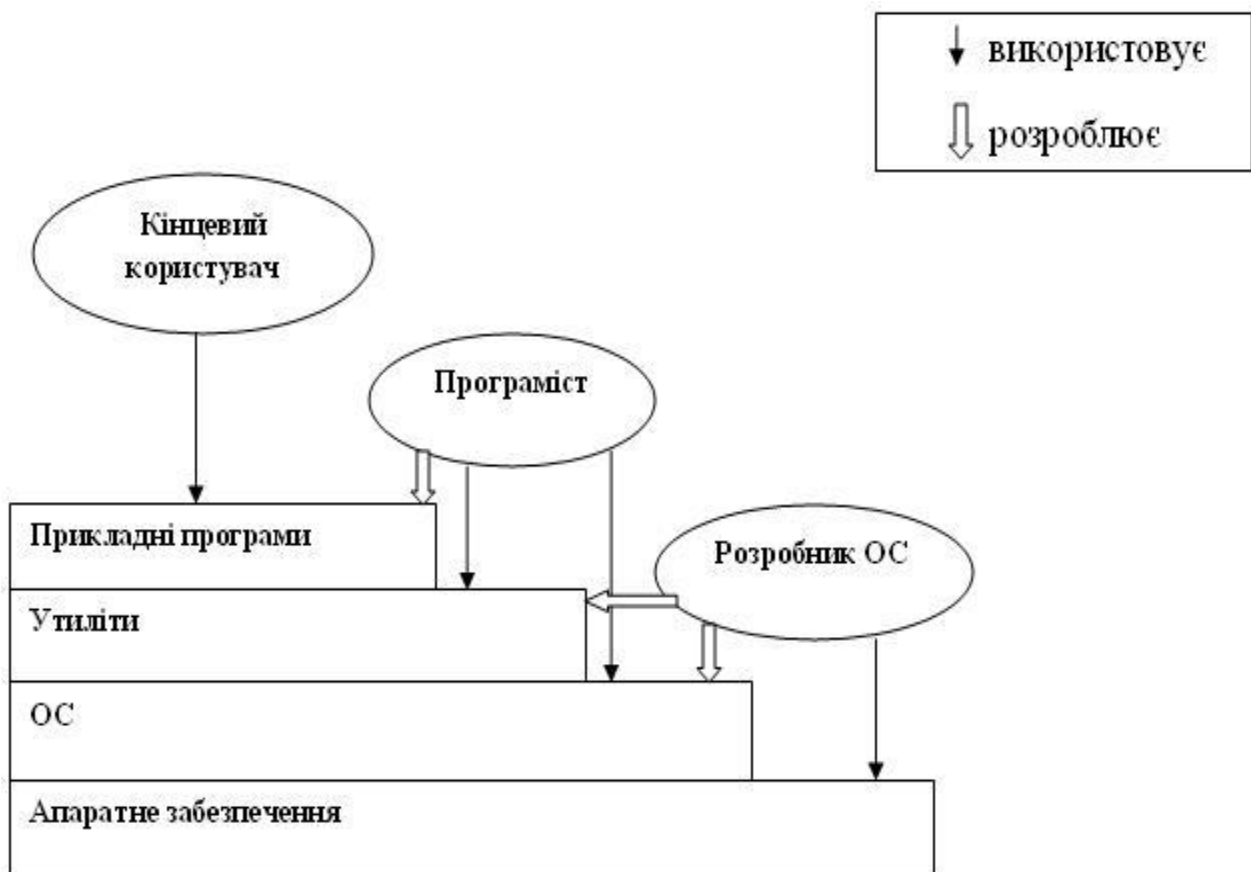


Рисунок 1.1 – Ієрархічна структура програмного забезпечення

На рисунку 1.2 подано більш докладну схему ієрархічної структури програмно-апаратного забезпечення комп'ютера.

Самий нижній рівень містить різні пристрої комп'ютера, що складаються мікросхем, провідників, джерел живлення і т.п. Цей рівень можна розділити на підрівні, наприклад контролери пристроїв, а потім самі пристрої. Можливий розподіл і на більше число рівнів.

Вище розташований мікроархітектурний рівень, на якому фізичні пристрої розглядаються як окремі функціональні одиниці.

На мікроархітектурному рівні знаходяться внутрішні реєстри центрального процесора (їх може бути декілька) і арифметико-логічні пристрої (АЛП) з засобами управління ними. На цьому рівні реалізується виконання машинних команд. У процесі виконання команд використовуються реєстри процесора і пристроїв, а також інші можливості апаратури.

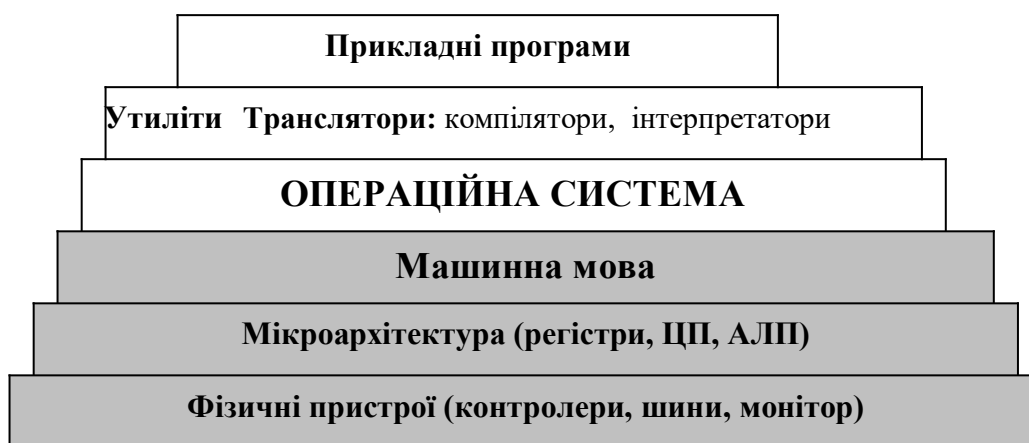




Рисунок 1.2 – Ієрархічна структура програмно-апаратних засобів комп'ютера:

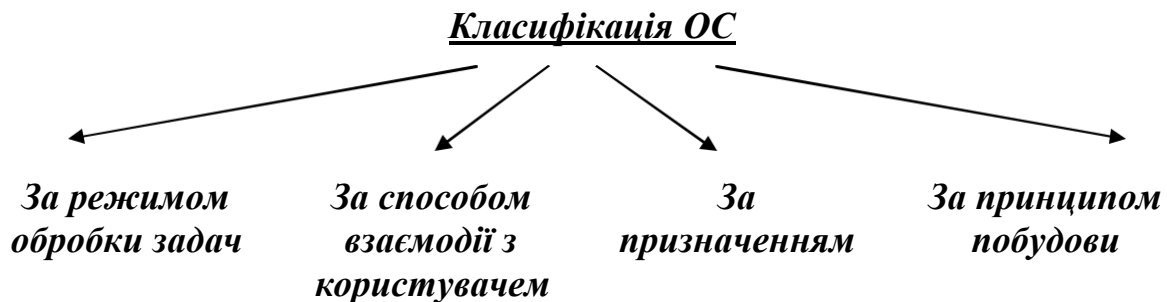
-  - рівні доступні для розробників ОС;
-  - рівні доступні для користувача і програміста.

Команди, видимі для працюючого на асемблері програміста, формують рівень ISA (Instruction Set Architecture – архітектура системи команд), так звана машинна мова.

Операційна система призначена для того, щоб приховати всі ці складності. Таким чином, операційна система – це набір програм, які контролюють роботу прикладних програм і системних додатків, і виконує роль інтерфейсу між користувачами, програмістами, прикладними програмами, системними додатками і апаратним забезпеченням комп'ютера.

## Сервіси, що надають типові операційні системи:

- Розробка програм.
- Виконання програм.
- Доступ до пристроїв введення-виведення.
- Доступ до файлів.
- Розподіл ресурсів.
- Визначення і обробка помилок.
- Облік використання ресурсів.
- Засоби комунікації.
- Захист даних і адміністрування.



### *За режимом обробки задач.*

- Однопрограмні (наприклад, MS-DOS).
- Мультипрограмні (більшість сучасних ОС).

### *За способом взаємодії з користувачем.*

- Діалогові ОС.
- Системи пакетної обробки.

### *За призначенням.*

*ОС мейнфреймів* (орієнтовані на одночасну обробку великої кількості завдань, що потребують багато операцій введення/виведення; можуть виконувати три види обслуговування (часто – всі разом): пакетну обробку, обробку транзакцій і роботу в режимі поділу часу. Приклад: OS/390, що створена на основі OS/360).

*Серверні ОС* (UNIX, Windows 2000, Linux).

*Мережеві ОС* (при роботі з такою ОС користувачі знають про існування в мережі безлічі ЕОМ, можуть реєструватися на віддалених

комп'ютерах і копіювати файли з однієї ЕОМ на іншу. Кожен комп'ютер при цьому працює під управлінням локальної ОС і має свого власного локального користувача (користувачів)).

*Розподілені ОС* (розподілені ОС повинні мати в своєму складі спеціальні програми для оптимізації розпаралелювання задач. Приклади розподілених ОС: Mach, Chorus, Sprite).

*ОС для персональних комп'ютерів* (Windows, Mac OS, Linux).

*ОС реального часу* (обробляє завдання, що надходять протягом заданих інтервалів часу, які не можна перевищувати; використовуються для керування технологічними процесами і лабораторними експериментами, в роботах систем управління повітряним рухом, в телекомунікаціях).

*Вбудовані ОС* (для забезпечення роботи мікрокомп'ютерів, вбудованих в різні побутові пристрої, мають такі ж характеристики, що і ОС реального часу, повинні займати невелике місце в пам'яті з урахуванням обмеження по обчислювальній потужності мікрокомп'ютерів. Приклади вбудованих ОС: PalmOS, Windows CE.

*За принципом побудови.*

Макроядерні (Монолітні).

Мікроядерні.

Ієрархічні.

## 1.2 Структура операційних систем

Для більш докладного роз'яснення класифікації ОС за принципом побудови з'ясуємо з чого складається операційна система.

**До складу операційної системи входять:**

*ядро операційної системи*, що забезпечує розподіл та управління ресурсами обчислювальної системи;

*базовий набір прикладного програмного забезпечення;*

*системні бібліотеки та програми обслуговування.*

**Ядро операційної системи** — це набір функцій, структур даних та окремих програмних модулів, які завантажуються в пам'ять комп'ютера при завантаженні операційної системи та забезпечують три типи системних сервісів:

- управління вводом-виводом інформації (підсистема введення/виведення ОС);
- управління оперативною пам'яттю (підсистема управління оперативною пам'яттю ОС);
- управління процесами (підсистема управління процесами ОС).

Типова ОС складається з таких компонентів:

- керування процесами;
- керування пам'яттю;
- підсистема вводу - виводу;
- файлова система;
- підтримка мережі;
- користувацький інтерфейс.

***Структура ОС визначає, як пов'язані між собою її компоненти.***

***Монолітна структура (структура макроядра).*** Всі програми ОС монолітної структури зібрані в один модуль. Така ОС являє собою набір процедур, кожна з яких при необхідності може викликати інші процедури, так як в монолітній ОС всі процедури відтрансльовані і скомпоновані в єдиний виконуваний файл. У порівнянні з іншими структурами, монолітну ОС важко розробляти, відлагоджувати, розширювати, і як наслідок у неї низька надійність, але більш висока швидкодія. Приклади таких ОС: MS-DOS, ранній Unix.

***Ієрархічна структура (багат шарове ядро).*** Основні принципи побудови ОС ієрархічної структури:

- кожний рівень контролює свій ряд структур даних і надає функції доступу до них;
- кожний рівень нічого «не знає» про структуру рівнів, що знаходяться нижче, доступ до сервісів, яких відбувається через певний інтерфейс;
- кожний рівень нічого «не знає» про існування рівнів, що знаходяться вище.

Переваги ієрархічної ОС: модульність, простота відладки і підтримки, можливість заміни або модифікації окремого шару.

**Структура мікроядра.** В такій ОС ядро виконує невелику кількість самих важливих функцій: робота з адресним простором, міжпроцесна взаємодія, основне планування. Інші функції ОС виконуються процесами або серверами. Ці процеси запускаються в режимі користувача, а мікроядро працює ними таким же чином, як і з іншими додатками. При цьому спрощується розробка ОС, підвищується її гнучкість, але разом з тим відбувається зменшення швидкодії обчислювальних процесів.