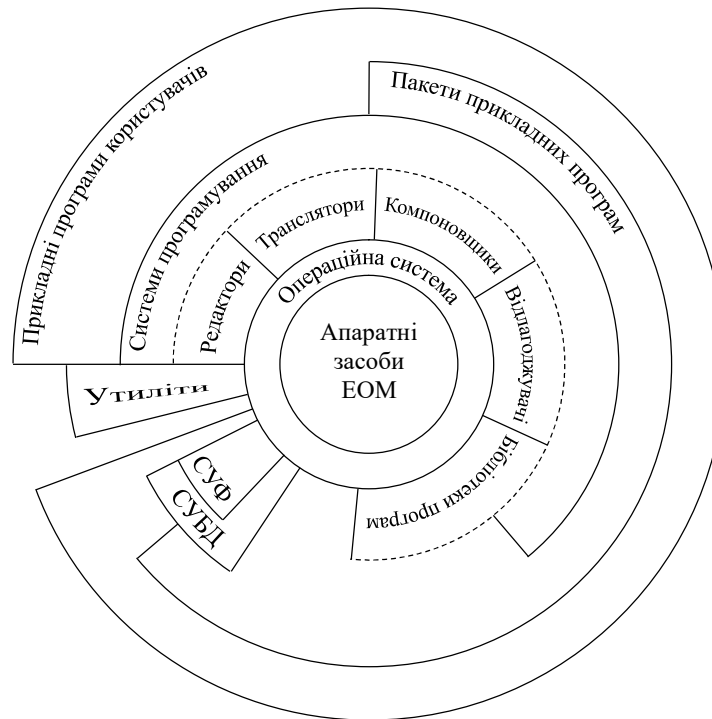


# Системне програмне забезпечення

## Тема 1: Вступ. Мета та задачі курсу.

System Software – означає програми та комплекси програм, які є спільними для всіх, хто використовує технічні засоби комп'ютера і які застосовуються, як для автоматизації розробки (створення) нових програм, так і для організації виконання існуючих програм. З цих позицій системне програмне забезпечення (СПЗ) може бути розділене на такі групи:

1. Операційні системи.
2. Системи програмування.
3. Утиліти.



Узагальнена структура програмного забезпечення обчислювальної системи

Під ОС звичайно розуміють комплекс керуючих та обробляючих програм, який з одного боку виступає як інтерфейс між апаратурою комп'ютера та користувачем з його задачами, а з другого боку – призначений для найбільш ефективного використання ресурсів комп'ютера (або ширше – обчислюваної системи) та організації надійних обчислень.

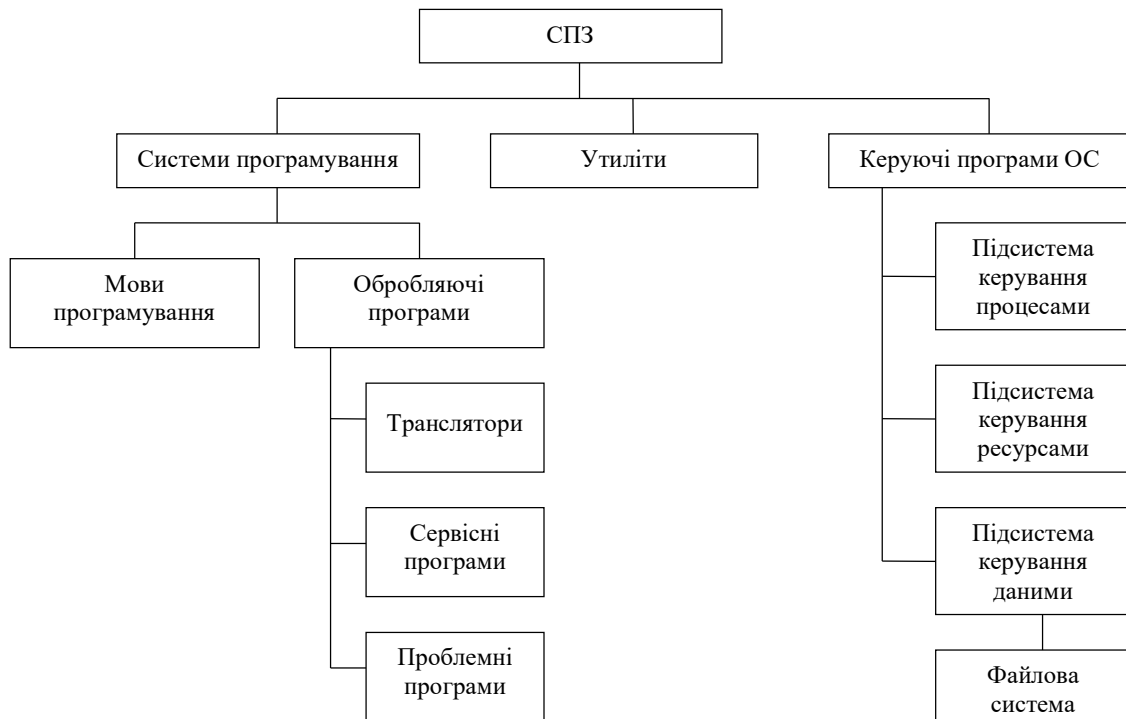
Довільний з компонентів прикладного програмування обов'язково працює під керуванням ОС. Жодна з компонент програмного забезпечення, за виключенням ОС не має безпосереднього доступу до апаратури комп'ютера. Навіть користувачі взаємодіють із своїми програмами через інтерфейс ОС. Довільні команди, перш ніж потрапити у прикладу програму, спочатку проходять через ОС.

Системи програмування включають в себе перш за все такі компоненти як транслятори з мов програмування, бібліотеки підпрограм, редактори, компоновщики та відлагоджувачі. Зовсім самостійних, тобто відірваних від конкретної ОС систем програмування не буває. Довільна система програмування може працювати тільки під відповідною ОС, під яку вона і

створена. Але система програмування може давати можливість розробляти програмне забезпечення і під інші ОС. У такому випадку, коли програми, що створюється, повинні працювати на іншій апаратній базі, маємо крос – систему.

Утиліти. Це спеціальні системні програми за допомогою яких можна як обслуговувати саму ОС, так і готувати для роботи носії даних, виконувати перекодування даних, оптимізацію розміщення даних на носії та виконувати деякі інші роботи, пов'язані із обслуговуванням обчислювальних систем. Зрозуміло, що утиліти можуть працювати тільки під відповідною операційною системою.

Таким чином, типова структура СПЗ має такий вигляд (Рис. \_\_)



Типова структура системного програмного забезпечення (СПЗ)

Основними функціями ОС є:

1. Сприймання від користувача (або від оператора системи) завдань або команд у вигляді директив (або команд) оператора, а також у вигляді вказівок від маніпуляторів (наприклад, миші) та їх обробка.

2. Завантаження, запуск, призупинення, зупинка програм.

Керування файловими системами та системами управління базами даних.

3. Забезпечення режиму мультипрограмування, тобто виконання двох та більше програм на одному процесорі.

4. Забезпечення функцій з організації та керування усіма операціями вводу/виводу.

5. Забезпечення виконання обмежень на час відповіді в режимі реального часу (для ОС відповідного типу).

6. Розподіл пам'яті (як основної, так і зовнішньої), та організація віртуальної пам'яті.

7. Планування та диспетчеризація задач у відповідності із заданими стратегією та дисципліною обслуговування.

8. Організація механізмів обміну повідомленнями і даними між програмами, що виконуються, а також захист одної програми від впливу інших.

9. Забезпечення роботи систем програмування.

Таким чином, ОС виконує функції керування обчислювальними процесами в комп'ютерній системі, розподіляє ресурси комп'ютерної системи між різними обчислювальними процесами і створює програмне середовище, в якому виконуються прикладні програми користувачів. Таке середовище називається операційним середовищем або операційним оточенням.

Взаємодія користувача з комп'ютерною системою відбувається через інтерфейс користувача, який підтримується ОС. Надалі приймається, що функції ОС реалізуються тільки програмним способом. Але деякі з них можуть бути представлені програмно-апаратними засобами (firmware) у вигляді постійних запам'ятовуючих пристроїв (BIOS).

Операційна система складається з деякої множини стандартних сервісних програм, які можуть бути використані в процесі виконання задачі і які надають засоби для керування ресурсами обчислювальної системи, виділяючи їх користувачам по потребі.

Наприклад: користування вводом/виводом.

Операційне оточення разом з реальною машиною (апаратурою комп'ютера) створюють розширену машину, яка і використовується під час виконання програм. Завдяки операційному оточенню кожна програма (тобто завдання для ОС) виконується ніби на окремій розширеній машині, хоча в дійсності ресурси реальної машини можуть розподілятися між багатьма користувачами.

### Типи операційних систем

Не завжди можна досягнути повної ясності у визначенні типів ОС, які можуть підпадати більш ніж під одну категорію класифікації.

1. Класифікація, за кількістю користувачів, які одночасно обслуговуються ОС:

Одно програмна — забезпечує роботу однієї задачі одного користувача.

Мульти (багато) програмна — дозволяє одночасно виконувати декілька програм одного користувача.

Багатокористувацька — багато задач багатьох користувачів, поділяється на мульти-процесорну та мульти-машинну (мережева ОС).

Основна мета мультипрограмування — підвищення продуктивності обчислювальної системи за рахунок розподілення її ресурсів між декількома завданнями.

2. Класифікація за типами доступу, який надається інтерфейсом користувача.

Системи пакетної обробки – обробляють потік завдань. Все керування з читання та виконання завданнями бере на себе ОС. Зараз втратили актуальність.

Діалоговий або інтерактивний доступ – забезпечується для деякої кількості користувачів одночасно системами розподілу часу.

ОС виконує директиви користувачів в темпі надходження та намагається дати відповідь на кожну команду користувача за найкоротший час.

Для обробки зовнішніх сигналів, які надходять, наприклад, з різних давачів, і швидкої відповіді на них використовуються ОС реального часу. Такі ОС працюють на комп'ютерах, для яких час є критичним параметром. Вони повинні забезпечувати гарантований час реакції на зовнішні події.

Досить часто всі ознаки класифікації реалізуються в одній системі.

На рівні функцій та послуг, що надаються операційним оточенням, є багато спільного між ОС, які на рівні інтерфейсу користувача зовсім різні (і навпаки).

Операційне оточення може включати декілька інтерфейсів: користувацькі і програмні:

Linux – має інтерфейс командного рядка (shell);

інтерфейс Midnight Commander (типу NC);

графічні інтерфейси – X-Window з різними менеджерами вікон (KDE, Gnome)

Програмні інтерфейси – в Linux програми можуть звертатись як до ОС за відповідними сервісами і функціями, так і до графічної підсистеми (якщо вона використовується).

З точки зору процесора комп'ютера, як і всього комп'ютера в цілому, двійкова програма, створена для роботи в середовищі Linux, використовує ті самі команди і формати даних, що і програма, створена для роботи в середовищі Windows NT.

Але в першому випадку ми маємо звертання до одного операційного середовища, а в другому випадку – а в другому випадку – до другого. Програма, створена під Windows безпосередньо, не буде виконуватись в Linux. Але якщо в ОС Linux створити повноцінне операційне середовище Windows, то Windows-програма може бути виконана.

Таким чином операційне середовище – це системне програмне оточення, в якому можуть виконуватись програми, створені за правилами роботи цього середовища.

## Тема 2: Процеси.

### Керування процесами

Поняття „обчислювальний процес” є одним із основних при вивченні операційних систем. Притримуємось такого визначення: Процес (або задача) – це програма під час виконання на процесорі із послідовним виконанням команд. Сам процесор розглядається в двох аспектах:

1. Він є носієм даних;
2. Виконує операції, пов'язані з обробкою цих даних.

Процесом може бути:

- виконання утиліти;
- виконання прикладної програми;
- трансляція вихідної програми (одної програми – один процес, іншої програми – інший процес).

Розглядаємо комп'ютер з одним центральним процесором, але всі подальші міркування справедливі і для багатопроцесорних систем.



Основні стани процесу

За період свого існування процес приймає ряд дискретних станів. Зміну станів процесу можуть викликати різні події.

Процес знаходиться в стані виконання, якщо в біжучий момент йому надається центральний процесор (CPU).

Процес знаходиться в стані готовності, якщо він міг би одразу використати CPU, який знаходиться в його розпорядженні.