

Лекція 1

ПОНЯТТЯ СИСТЕМНОГО ПРОГРАМУВАННЯ

Сьогодні програмування є базовою, основною навичкою практично в будь-якій професії. З кожним наступним роком програмування буде ставати все більш і більш актуальним. Так, Вам треба навчитися програмувати, навіть якщо Ви не збираєтеся ставати програмістом. Програмування визначається як процес створення програмного забезпечення (ПЗ). Програмне забезпечення комп'ютера ділиться на наступні класи (рис 1.1.):

— системне ПЗ; — прикладне ПЗ; — системи програмування.

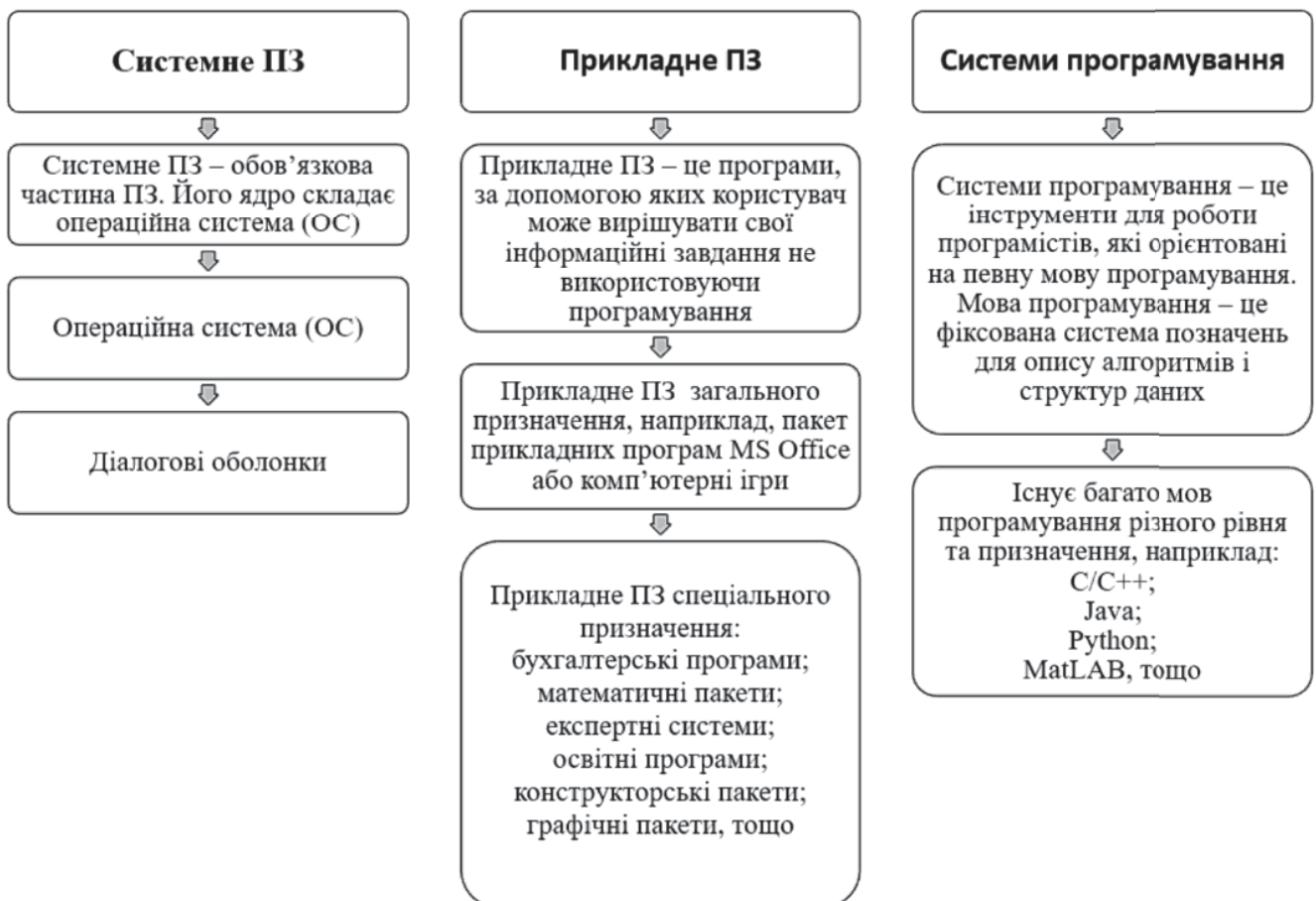


Рис.1.1. — Структура програмного забезпечення

Системне програмне забезпечення — комплекс програм, які забезпечують управління компонентами комп'ютерної системи, такими як процесор, оперативна пам'ять, пристрої введення-виведення, мережеве обладнання. Системне програмне забезпечення виступає як «міжшаровий інтерфейс», з одного боку якого апаратура, а з іншого — додатки користувача.

На відміну від прикладного програмного забезпечення, системне програмне забезпечення не вирішує конкретні практичні завдання, а лише забезпечує роботу інших програм, надаючи їм сервісні функції, що абстрагують деталі апаратної і мікропрограмної реалізації обчислювальної системи, управляють її апаратними ресурсами.

Системне програмування — створення системного програмного забезпечення.

Системний програміст — програміст, що спеціалізується на системному програмуванні.

Відзначимо, що системне програмування є досить специфічним для кожної операційної системи.

На початок 2019 року на ринку настільних комп'ютерів та ноутбуків ОС Windows є найбільш затребуваною операційною системою (табл. 1.1.).

Таблиця 1.1. — Частки операційних систем на ринку настільних комп'ютерів та ноутбуків

Windows	OS X	Unknown	Linux	Chrome OS	FreeBSD
75.47%	12.33%	9.41%	1.61%	1.17%	0.01%

Маємо відзначити, що системне програмування для OS X і Linux виглядає досить схожим на системне програмування для Windows, проте є трохи менш звичним і тому вимагає більше часу.

Однак, розуміючи основи, Ви завжди зможете розібратися в тій чи іншій операційній системі.

Будь-яку операційну систему можна представити у вигляді сукупності ядра системи, яке має доступ до апаратури і оперує файлами і процесами, і оболонки (командного інтерпретатора) з утилітами, які дозволяють користувачеві отримати доступ до функціональності ядра операційної системи (рис 1.2.).

Іншою важливою складовою операційної системи є драйвери.

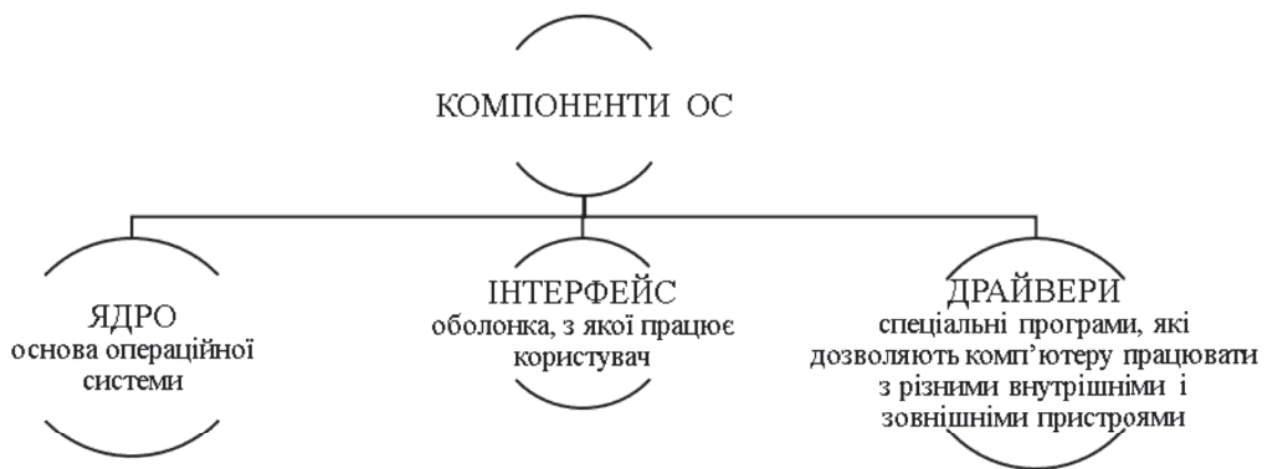


Рис. 1.2. — Склад ОС

Операційні системи не завжди мали графічний інтерфейс. Як приклад, досить згадати ОС MS-DOS. Стисла історія операційних систем від корпорації Microsoft наведена на рис. 1.3.

Графічні надбудови над DOS



Windows 1.0
(20 листопада
1985 р.)



Windows 2.0
(2 квітня 1987 р.)



Windows
3.0
(22
травня
1990 р.)



Windows 3.1
(11 листопада
1993 р.)

Сімейство 9x



Windows 95
(24 серпня 1995 р.)



Windows 98
(25 червня 1998 р.)



Windows ME
(14 вересня 2000 р.)

Сімейство Windows NT



Windows
NT 3.1
(27 липня
1993 р.)



Windows
2000
(17
лютого
2000 р.)



Windows
XP
(25
жовтня
2001 р.)



Windows
Vista
(30 січня
2007 р.)



Windows
7
(22
жовтня
2009 р.)



Windows
8
(26
жовтня
2012 р.)



Windows
10
(29 липня
2015 р.)

Рис. 1.3. — Історія ОС Windows

Графічний інтерфейс перших операційних систем був вельми примітивним (рис 1.4.).

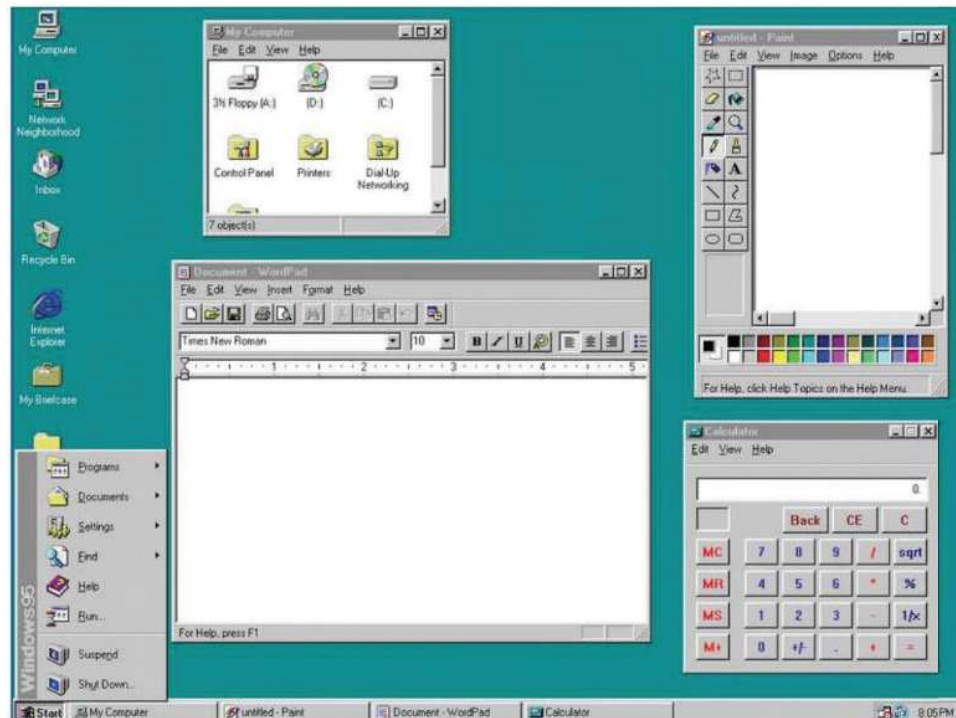


Рис. 1.4. — Графічний інтерфейс ОС Windows 95

До використання графічного інтерфейсу механізм взаємодії з користувачем був наступним: у відповідь на запрошення («підказку», `prompt`), що видається оболонкою (яка знаходиться у режимі очікування), користувач вводить деяку команду (функціональність цієї команди може бути реалізована або самою оболонкою, або певною зовнішньою утилітою), оболонка виконує її, при необхідності виводячи на екран деяку інформацію (рис. 1.5.)

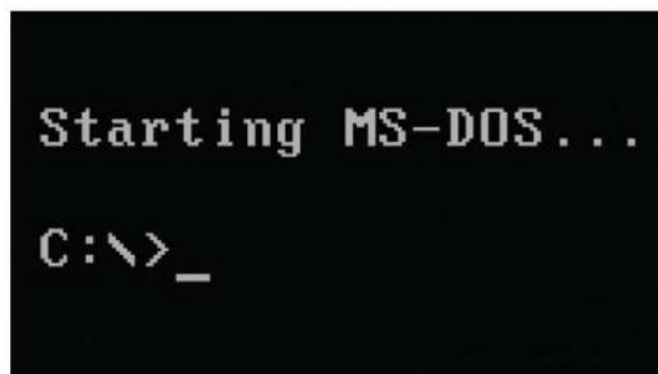


Рис. 1.5. — Приклад роботи в ОС MS-DOS

Такий вид інтерфейсу не залишився в минулому, а використовується професіоналами до цього дня, завдяки ряду переваг:

- значно розширюються функціональність операційної системи;
- виникає більше розуміння механізмів роботи операційної системи;
- значно спрощується автоматизація.

У всіх версіях операційної системи Windows підтримується інтерактивна оболонка командного рядка (command shell) і за замовчуванням встановлюється певний набір утиліт командного рядка (кількість і склад цих утиліт залежить від версії операційної системи).

Починаючи з версії Windows NT, оболонка командного рядка представляється інтерпретатором cmd.exe, який розширює можливості оболонки command.com операційної системи MS-DOS.

Наступним кроком у розвитку засобів і технологій автоматизації в операційній системі Windows стала поява сервера сценаріїв Windows Script Host (WSH, лістинг 1.1.).

```
Set WshShell=WScript.CreateObject("WScript.Shell")
WshShell.Run "",1,True
WshShell.Popup("")
WScript.Quit
```

Лістинг 1.1. — Приклад коду для WSH

Цей інструмент був розроблений для всіх версій Windows і дозволяє безпосередньо в операційній системі виконувати сценарії на повноцінних мовах сценаріїв (за замовчуванням, VBScript і JScript). VBScript і JScript — це повноцінні алгоритмічні мови, що мають вбудовані функції і методи для обробки символьних рядків, виконання математичних операцій, обробки виняткових ситуацій і т.д.

До початку XXI століття розробники Microsoft випустили нову оболонку PowerShell (рис. 1.6.), в якій були об'єднані всі кращі аспекти інших оболонок

командного рядка з різних операційних систем. За їх словами, сильний вплив на PowerShell здійснили такі продукти:

- BASH, KSH (конвеєризація або композиція команд);
- AS400 / VMS (стандартні назви команд, що прискорюють вивчення);
- TCL / WSH (підтримка можливості вбудовування і декількох мов);
- PERL, PYTHON (виразність і стиль).

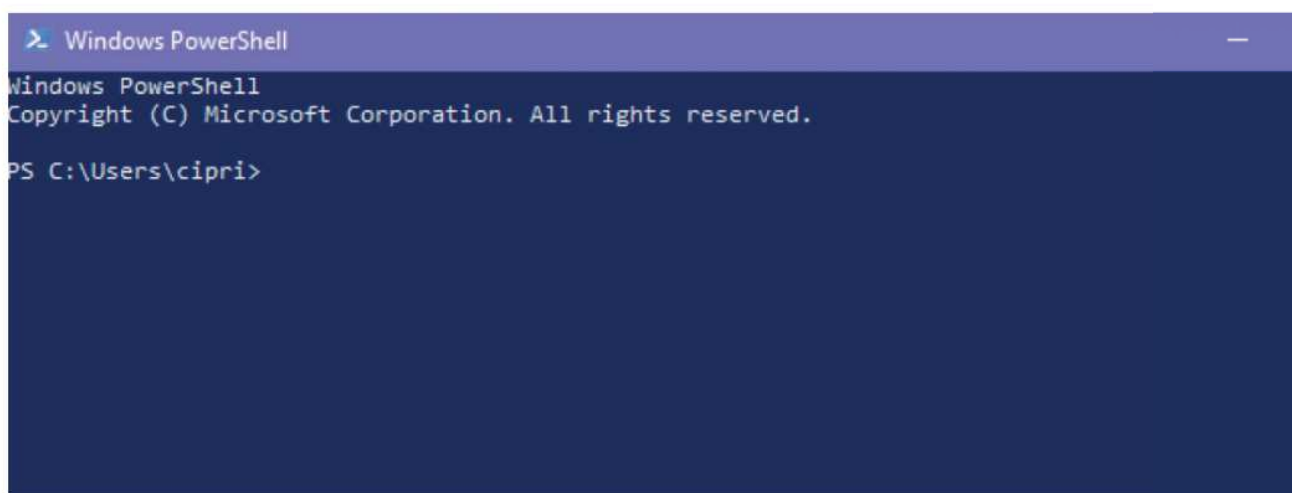


Рис. 1.6. — Приклад роботи в PowerShell

За традицією ми почнемо наш екскурс в системне програмування саме з командної оболонки cmd.

В операційній системі Windows, як і в інших операційних системах, інтерактивні (такі, що набираються з клавіатури і відразу ж виконуються) команди виконуються за допомогою так званого командного інтерпретатора, який інакше називається командним процесором або оболонкою командного рядка (command shell).

У Windows файл cmd.exe, як і інші виконувані файли (рис. 1.7.), що відповідають зовнішнім командам операційної системи, знаходяться в каталозі %SystemRoot%\SYSTEM32 (значенням змінної середовища %SystemRoot% є системний каталог Windows, зазвичай C:\Windows).

Рис 1.7. — Виконуваний файл cmd

Запустити командний інтерпретатор можна з меню «Пуск» або за допомогою утиліти «Виконати» (рис. 1.8.).

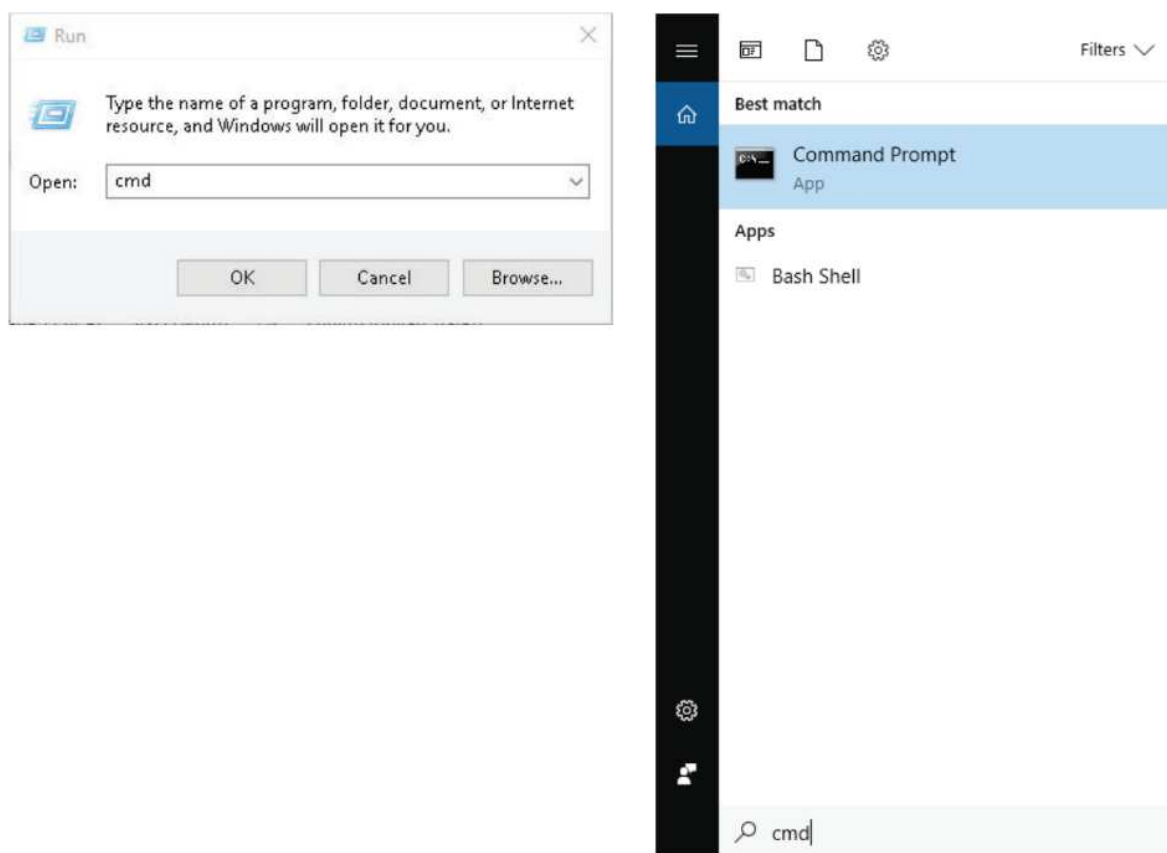


Рис. 1.8. — Приклад запуску командного інтерпретатора

Розглянемо структуру самого командного рядка і принцип роботи з ним (рис. 1.9.). Для того, щоб виконати команду, після запрошення командного рядка (наприклад, C:\>) Ви маєте ввести ім'я цієї команди (реєстр не важливий), її параметри і ключі (якщо вони необхідні) та натиснути клавішу «Enter».


```
D:\>COPY C:\myfile.txt A:\ /V
```

↑ ↑ ↑
ім'я команди параметри ключ

Рис. 1.9. — Структура команди командного рядка

Деякі команди розпізнаються і виконуються безпосередньо самим командним інтерпретатором — такі команди називаються внутрішніми (наприклад, COPY або DIR). Інші команди системи являють собою окремі програми, розташовані за замовчуванням в тому ж каталозі, що і cmd.exe, які Windows завантажує і виконує аналогічно іншим програмам. Такі команди називаються зовнішніми (наприклад, MORE або XCOPY, рис. 1.10.).



 more	4/12/2018 2:34 AM	MS-DOS Applicati...	28 KB
 xcopy	4/12/2018 2:34 AM	Application	47 KB

Рис. 1.10. — Виконувані файли команд MORE і XCOPY

За замовчуванням, при запуску інтерпретатора командного рядка ми знаходимося в основному каталозі поточного користувача, який відображається в запрошенні інтерпретатора (prompt).

Ми можемо легко змінити каталог за допомогою команди CD. Команда має наступний синтаксис:

CD [/D] [диск:] [шлях] [..]

Наприклад, перейдемо в папку Windows (лістинг 1.2.)

```
C:\>CD c:\Windows  
c:\Windows>
```

Лістинг 1.2. — Приклад переходу в директорію C:\Windows

Існують спеціальні форми запису команди CD, що полегшують перехід по директоріям файлової системи. Наприклад, для переходу в корінь диска застосовується команда CD \ (лістинг 1.3.).

```
c:\Windows>CD \  
c:\>
```

Лістинг 1.3. — Приклад переходу в кореневу директорію

Щоб перейти з поточного підкаталогу в батьківський каталог застосовується команда CD .. (лістинг 1.4.).

```
c:\>CD c:\Windows\Fonts  
c:\Windows\Fonts>CD..  
c:\Windows>CD ..  
c:\>
```

Лістинг 1.4. — Приклад переходу в батьківську директорію

Ключ /D в команді CD застосовується в тому випадку, коли необхідно змінити диск. Наприклад, спробуємо перейти на диск D, а потім на диск G (лістинг 1.5.).

```
c:\>CD /d d:\  
d:\>CD /d g:\  
g:\>
```

Лістинг 1.5. — Приклад переходу на інший диск

Зауважимо, що існує ще один більш простий спосіб переходу з одного диска на інший.

Для цього просто достатньо ввести букву диска на який ми хочемо перейти і двокрапку після неї (лістинг 1.6.). Наприклад, перейдемо на диск D

```
c:\>d:  
d:\>
```

Лістинг 1.6. — Приклад переходу на інший диск без команди CD

Іншою найважливішою командою є команда DIR, що призначена для перегляду вмісту директорії. Команда DIR використовується для виведення на екран інформації про вміст каталогів і дисків. Дана команда має наступний синтаксис:

```
DIR [диск:][шлях][ім'я_файла] [/A[:]атрибути] [/B] [/C] [/D] [/L]  
[/N]/[O[:]порядок сортування] [/P] [/Q] [/R] [/S] [/T[:]час] [/W] [/X]  
[/4][диск:][шлях][ім'я_файла].
```

Як видно з синтаксису, команду DIR можна використовувати без параметрів. В цьому випадку команда DIR діє на поточний каталог. На екран виводиться мітка диска (тома), його серійний номер, назви файлів і підкаталогів, що знаходяться в поточному каталозі, і дата їх останньої зміни. Для файлів вказується їх розмір, загальна кількість. Також вказується загальна кількість папок та обсяг вільного місця на диску. Для прикладу, можна застосувати команду DIR для диска C: (лістинг 1.7.).

```
c:\>DIR  
Volume in drive C has no label.  
Volume Serial Number is BA03-7F8D  
  
Directory of c:\  
02/18/2018  03:51 PM    <DIR>          Intel  
03/13/2018  12:52 AM    <DIR>          Octave  
04/12/2018  02:38 AM    <DIR>          PerfLogs  
04/25/2019  10:51 AM    <DIR>          Program Files  
05/16/2019  08:55 AM    <DIR>          Program Files (x86)  
09/19/2018  03:03 PM    <DIR>          Users  
08/04/2018  04:31 PM    <DIR>          WCH.CN  
05/13/2019  01:03 PM    <DIR>          Windows  
               0 File(s)                0 bytes  
               11 Dir(s)  55,998,435,328 bytes free
```

Лістинг 1.7. — Приклад застосування команди DIR

Команда DIR має досить велику кількість різних ключів. Розглянемо деякі з них.

Ключ /W використовується в тому випадку, якщо необхідно вивести список файлів і каталогів в одному рядку з максимально можливою їх кількістю (лістинг 1.8.), що уміщається на екрані (широкий формат). Файли і папки виводяться в алфавітному порядку (упорядковано за строками).

```
c:\>DIR /w
Volume in drive C has no label.
Volume Serial Number is BA03-7F8D

Directory of c:\

[1]                [Intel]
[Octave]            [PerfLogs]
[Program Files]     [Program Files (x86)]
[Users]             [VAGTool]
[VCDS RUS 14.10.0]  [WCH.CN]
[Windows]

                0 File(s)                0 bytes
                11 Dir(s)  55,995,498,496 bytes free
```

Лістинг 1.8. — Приклад використання команди DIR з ключем /W

Ключ /A:[атрибути] застосовується для виведення інформації про файли і папки, які мають вказані атрибути (лістинг 1.9.). Атрибутами можуть бути:

- Н — на екран виводяться приховані файли і папки;
- S — на екран виводяться системні файли і папки;
- R — на екран виводяться файли і папки з атрибутом «лише для читання»;
- A — на екран виводяться файли і папки з встановленим архівним атрибутом;
- I — на екран виводяться файли і папки з неіндексованим вмістом.

Ключ /D забезпечує виведення імен файлів по стовпцях (лістинг 1.9.).

```

c:\Windows>DIR /A:R /D
Volume in drive C has no label.
Volume Serial Number is BA03-7F8D
Directory of c:\Windows

[assembly]                [Offline Web Pages]
[Fonts]                    [PrintDialog]
[ImmersiveControlPanel] WindowsShell.Manifest
[media]
[Microsoft.NET]
                1 File(s)                670 bytes
                7 Dir(s)  55,997,251,584 bytes free

```

Лістинг 1.9. — Приклад використання команди DIR з ключем /A і /D

Ключ /O:[сортування] задає порядок сортування папок і файлів при виведенні на екран. Існує кілька видів сортування при використанні команди DIR з ключем /O:

- /O:N — сортування по імені каталога або файла (за алфавітом);
- /O:E — сортування за розширенням (за алфавітом);
- /O:D — сортування за датою і часом (починаючи з самого старого).

При цьому префікс «-» інвертує порядок.

Ключ /Q дозволяє виводити інформацію про власника даного файлу або папки (лістинг 1.10.). Наприклад, команда:

DIR d:\folder /q

виводить на екран вміст папки «FOLDER» із зазначенням власника всіх папок і файлів.

```

c:\>DIR /Q
Volume in drive C has no label.
Volume Serial Number is BA03-7F8D

Directory of c:\

```

Лістинг 1.10. — Приклад використання команди DIR з ключем /Q


```

03/05/2019  11:13 AM    <DIR>  DESKTOP-1Q6B8MD\Sandy-l1
02/18/2018  03:51 PM    <DIR>  BUILTIN\Администраторы Intel
03/13/2018  12:52 AM    <DIR>  BUILTIN\Администраторы Octave
04/12/2018  02:38 AM    <DIR>  ...                               PerfLogs
04/25/2019  10:51 AM    <DIR>  NT SERVICE\TrustedInstaProgram Files
05/16/2019  08:55 AM    <DIR>  NT SERVICE\TrustedInstaProgram Files (x86)
09/19/2018  03:03 PM    <DIR>  NT AUTHORITY\СИСТЕМА  Users
11/21/2018  05:45 PM    <DIR>  BUILTIN\Администраторы VAGTool
08/04/2018  04:27 PM    <DIR>  BUILTIN\Администраторы VCDs RUS 14.10.0
08/04/2018  04:31 PM    <DIR>  DESKTOP-1Q6B8MD\Sandy-lwCH.CN
05/13/2019  01:03 PM    <DIR>  NT SERVICE\TrustedInstaWindows
      0 File(s)                0 bytes
     11 Dir(s)  55,994,294,272 bytes free

```

Лістинг 1.10. (закінчення) — Приклад використання команди DIR з ключем /Q

Взагалі кажучи, багато команд Windows мають велику кількість додаткових параметрів і ключів, запам'ятати які часто буває важко. Більшість команд забезпечено вбудованою довідкою, в якій коротко описуються призначення і синтаксис даної команди. Отримати доступ до такої довідки можна шляхом введення команди з ключем /? (лістинг 1.11.).

```

c:\>DIR /?
Displays a list of files and subdirectories in a directory.
DIR [drive:][path][filename] [/A[[:]attributes]] [/B] [/C] [/D] [/L] [/N]
  [/O[[:]sortorder]] [/P] [/Q] [/R] [/S] [/T[[:]timefield]] [/W] [/X]
[/4]
  [drive:][path][filename]
      Specifies drive, directory, and/or files to list.
/A    Displays files with specified attributes.
      attributes  D Directories      R Read-only files
                  H Hidden files    A Files ready for archiving
                  S System files    I Not content indexed files
                  L Reparse Points - Prefix meaning not
/B    Uses bare format (no heading information or summary).
/C    Display the thousand separator in file sizes. This is the default.
      Use /-C to disable display of separator.
/D    Same as wide but files are list sorted by column.

```

Лістинг 1.11. — Приклад використання команди DIR з ключем /?

Завдання. Вивести на екран список каталогів, що містяться на диску D:\. Файли повинні бути відсортовані за датою останньої зміни.

Як Ви могли помітити команди командного рядка за замовчуванням виводять інформацію про свою роботу у вікно середовища інтерпретатора командного рядка.

Однак, стандартний потік виведення інформації цілком можливо перенаправити. Наприклад, нехай ми хочемо вивести інформацію про вміст диска C:\ не на екран, а у файл. У цьому випадку використовується символ перенаправлення потоку > (лістинг 1.12.).

```
d:\3>DIR > aboutDIR.txt
d:\3>MORE aboutDIR.txt
Volume in drive D has no label.
Volume Serial Number is 2293-2D78

Directory of d:\3
05/30/2019  08:36 AM    <DIR>          .
05/30/2019  08:36 AM    <DIR>          ..
03/05/2019  11:19 AM                8 1.txt
03/05/2019  11:19 AM               10 2.txt
03/05/2019  11:19 AM               15 3.txt
03/05/2019  11:30 AM               34 5.txt
05/30/2019  08:36 AM                0 aboutDIR.txt
                    5 File(s)                67 bytes
                    2 Dir(s)  788,302,712,832 bytes free
```

Лістинг 1.12. — Приклад перенаправлення потоку виведення та його читання за допомогою команди MORE

Відзначимо, що символ перенаправлення потоку > створює новий файл, якщо він був відсутній раніше. Якщо цей файл існував раніше в цьому каталозі, він буде перезаписаний. Якщо потрібно дописати інформацію в кінець існуючого файлу, необхідно використовувати конструкцію:

команда >> ім'я_файлу

Наведемо приклад виконання команди, перенаправивши її потік виведення у файл з дозаписом в кінець файлу (лістинг 1.13.).

```
d:\3>DIR c:\ >> aboutDIR.txt
d:\3>MORE aboutDIR.txt
Volume in drive D has no label.
Volume Serial Number is 2293-2D78

Directory of d:\3
05/30/2019  08:36 AM    <DIR>          .
05/30/2019  08:36 AM    <DIR>          ..
03/05/2019  11:19 AM                8 1.txt
03/05/2019  11:19 AM               10 2.txt
03/05/2019  11:19 AM               15 3.txt
03/05/2019  11:30 AM               34 5.txt
05/30/2019  08:36 AM                0 aboutDIR.txt
                5 File(s)                67 bytes
                2 Dir(s)  788,302,712,832 bytes free
Volume in drive C has no label.
Volume Serial Number is BA03-7F8D

Directory of c:\
03/05/2019  11:13 AM    <DIR>          1
02/18/2018  03:51 PM    <DIR>          Intel
03/13/2018  12:52 AM    <DIR>          Octave
04/12/2018  02:38 AM    <DIR>          PerfLogs
04/25/2019  10:51 AM    <DIR>          Program Files
05/16/2019  08:55 AM    <DIR>          Program Files (x86)
09/19/2018  03:03 PM    <DIR>          Users
08/04/2018  04:31 PM    <DIR>          WCH.CN
05/13/2019  01:03 PM    <DIR>          Windows
                0 File(s)                0 bytes
               11 Dir(s)  55,987,040,256 bytes free
```

Вирішуване також і зворотнє завдання. Команди цілком можуть приймати інформацію не з клавіатури, а з файлів. Як приклад, можна навести команду SORT, яка здійснює сортування в порядку зростання. Наприклад, створимо текстовий файл з декількома словами (рис. 1.11.).

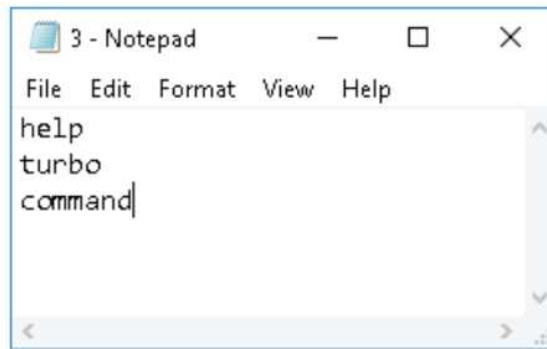


Рис. 1.11. — Вміст текстового файлу з послідовністю слів

Для того, щоб подати на вхід команди SORT вміст цього файлу використовується символ перенаправлення потоку < (лістинг 1.14.).

```
d:\3>SORT < 3.txt  
command  
help  
turbo
```

Лістинг 1.14. — Приклад перенаправлення потоку введення

Взагалі кажучи, в системі Windows існує 3 стандартних потоки:

1. Потік виведення (1).
2. Потік виведення помилок (2).
3. Потік введення (3).

Коли ми використовуємо символи >, >> і <, ми займаємося перенаправленням потоку виведення або потоку введення.

Наприклад, перенаправимо потік виведення команди DIR, але виклинемо помилку при її виконанні (лістинг 1.15.).

```
d:\>DIR d:\non_existing_folder > d:\1.txt  
File Not Found
```

Лістинг 1.15. — Спроба перенаправлення потоку виведення помилки за допомогою символу перенаправлення стандартного потоку виведення

Як бачимо, помилка все одно з'явилася на екрані. Для того, щоб перенаправити потік виведення помилок, досить поставити цифру 2 перед символом перенаправлення (лістинг 1.16.).

```
d:\>DIR d:\non_existing_folder 2> d:\1.txt  
Volume in drive D has no label.  
Volume Serial Number is 2293-2D78  
Directory of d:\  
d:\>MORE 1.txt  
File Not Found
```

Лістинг 1.16. — Приклад перенаправлення потоку виведення помилок

Відзначимо, що якби при виконанні команди не виникла б помилка, результат виконання був би виведений на екран. При цьому на практиці потік виведення і потік виведення помилок часто об'єднують в єдиний потік за допомогою наступної конструкції:

2> & 1

При цьому і в разі помилки, і в разі коректного виконання, вихідна інформація все одно потрапляє в файл (лістинг 1.17.).

```
d:\>DIR d:\1 > d:\out.txt 2>&1
```

Лістинг 1.17. — Приклад об'єднання стандартного потоку виведення і потоку виведення помилок і їх перенаправлення в файл

Символом конвеєризації є символ потоку |. Наприклад, ми можемо перенаправити вихід команди DIR на вхід команди SORT. В цьому випадку,

виведений на екран вміст директорії буде відсортовано за першою колонкою — за датою створення файлу або директорії (лістинг 1.18.).

```
d:\3>DIR | SORT
                2 Dir(s)  788,302,704,640 bytes free
                5 File(s)                1,324 bytes
Directory of d:\3
Volume in drive D has no label.
Volume Serial Number is 2293-2D78
03/05/2019  11:19 AM                8 1.txt
03/05/2019  11:19 AM               10 2.txt
05/30/2019  08:41 AM               20 3.txt
```

Лістинг 1.18. — Приклад використання символу конвеєризації

Домашнє завдання до Лекції 1.

1. Записати команди, які виводять в три окремі текстові файли:

— список прихованих файлів в директорії C:\Windows;

— список системних файлів в директорії C:\Windows;

— список файлів в директорії C:\Windows, які доступні тільки

для читання.

2. Записати методи копіювання інформації з командного рядка.