

1. Варіант № 27

1.1. Поясніть поняття «пріоритет процесу». Розгляньте роль, яку пріоритет процесу відіграє в операційній системі

Пріоритет процесу — це значення, яке показує, в якому порядку даний процес буде виконаний в черзі задач. В операційній системі пріоритети процесів відіграють значну роль у плануванні задач: вони допомагають планувальнику задач визначити, в якому порядку запускати процеси і як розподілити між ними ресурси залежно від того, які пріоритети мають дані процеси.

1.2. Що таке «право на читання», «право на запис», «право на виконання» по відношенню до файлів?

Відносно файлів права визначаються так:

1. Читання (r, read) — це право переглядати зміст файлу.
2. Запис (w, write) — це право змінювати зміст файлу, тобто редагувати, додавати або видаляти інформацію з нього.
3. Виконання (x, execute) — це право запускати файл на виконання, тобто запускати сценарій (скрипт) на обробку відповідним інтерпретатором або програму чи інший виконуваний файл — безпосередньо операційною системою.

1.3. Наведіть відомі вам команди для роботи з файлами і каталогами в операційних системах DOS, Windows, Linux, Unix. Дайте їм коротку характеристику. Наведіть приклади

Табл. 1: Команди для роботи з файлами і директоріями в операційних системах Unix і Linux

Команда	Призначення	Приклад викор.	Опис
cat	Вивід змісту файлів	cat file	Переглянути зміст файлу file
ls	Вивід змісту директорій	ls -l .	Переглянути зміст поточної директорії

Команда	Призначення	Приклад викор.	Опис
<code>touch</code>	Створення файлів	<code>touch file1</code>	Створити файл <code>file1</code>
<code>mkdir</code>	Створення директорій	<code>mkdir dir1</code>	Створити директорію <code>dir1</code>
<code>cp</code>	Копіювання файлів і директорій	<code>cp -r dir1 dir2</code>	Скопіювати директорію <code>dir1</code> і її зміст у директорію <code>dir2</code>
<code>mv</code>	Переміщення файлів і директорій	<code>mv dir2 dir3</code>	Перейменувати директорію <code>dir2</code> в <code>dir3</code>
<code>rm</code>	Видалення файлів і директорій	<code>rm -r dir1</code>	Видалити директорію <code>dir1</code>
<code>chown</code>	Зміна власника файлів або директорій	<code>chown user file1</code>	Призначити користувача <code>user</code> власником файлу <code>file1</code>
<code>chgrp</code>	Зміна групи, якій належить файл	<code>chgrp group file1</code>	Встановити, що групі <code>group</code> належить файл <code>file1</code>
<code>chmod</code>	Зміна режиму файлу	<code>chmod g+w file1</code>	Надати групі, якій належить файл <code>file1</code> , право на запис у нього

Табл. 2: Команди для роботи з файлами і директоріями в операційних системах DOS і Windows

Команда	Призначення	Приклад викор.	Опис
<code>type</code>	Вивід змісту файлів	<code>type file</code>	Переглянути зміст файлу <code>file</code>
<code>dir</code>	Вивід змісту директорій	<code>dir /b .</code>	Перегляд змісту поточної директорії (аналог <code>ls</code>)

Команда	Призначення	Приклад викор.	Опис
md	Створення директорій	md dir1	Створити директорію dir1
copy	Копіювання файлів	copy file1 file2	Скопіювати файл file1 у файл file2
xcopy	Копіювання директорій	xcopy dir1 dir2	Скопіювати директорію file1 у директорію file2
move	Переміщення файлів і директорій	move dir1 dir2	Перейменувати директорію copied-dir в ren-dir
del	Видалення файлів і директорій	del /S /Q dir1	Видалити директорію dir1
attrib	Зміна атрибутів файлів або директорій	attrib +R file1	Встановити атрибут «тільки для читання» файлу file1

1.3.1. Команди для роботи з файлами і директоріями в операційних системах Unix і Linux

Перегляд змісту

Щоб переглянути зміст файлу в операційних системах Unix і Linux використовують команду `cat`. Наприклад, щоб переглянути повний зміст файлу `file`, команду використовують так:

```
cat file
```

Взагалі, команда `cat` використовується для конкатенації змісту файлів або потоків, але без параметрів на одному вхідному файлі вона виводить зміст заданого файлу.

Також можна переглянути певну кількість перших або останніх рядків файлу за допомогою команд `head` і `tail` відповідно:

```
head file
```

```
tail file
```

Щоб переглянути зміст директорії, використовують команду `ls`:

```
ls
```

Також можна переглянути зміст і розгорнуті деталі про кожен з елементів директорії за допомогою параметра `-l`:

```
ls -l
```

Створення

Щоб створити файл виключно за допомогою команд, використовують команду `touch`. Вона змінює дату і час останнього звернення до заданого файлу, якщо він існує; якщо ж файл не існує, вона створює заданий файл. Наприклад, щоб створити файл `new-file` команда використовується так:

```
touch new-file
```

Щоб створити директорію, використовують команду `mkdir`:

```
mkdir new-dir
```

Часто необхідно створити цілу низку директорій, вкладених одна в одну. Для цього команду `mkdir` використовують з параметром `-p`. Наприклад:

```
mkdir -p new-dir/level-1/level-2/level-3
```

Копіювання

Щоб скопіювати файл або директорію, використовують команду `cp`. Наприклад, щоб скопіювати файл `new-file` у файл `copied-file`, команду запускають так:

```
cp new-file copied-file
```

Якщо ж треба скопіювати директорію з її вмістом, команду запускають з параметром `-r`:

```
cp -r new-dir copied-dir
```

Видалення

Щоб видалити файл або директорію, використовують команду `rm`. Наприклад, щоб видалити файл `new-file` команда використовується так:

```
rm new-file
```

Щоб видалити директорію, цю ж команду запускають з параметром `-r`:

```
rm -r new-dir
```

Також можна запустити видалення файлів і директорій в інтерактивному режимі, щоб підтверджувати видалення кожного елементу. Для цього потрібен параметр `-i`:

```
rm -i new-dir/
```

Зміна параметрів

Щоб змінити параметри файлів і директорій, використовують команди `chown`, `chgrp` і `chmod`. Команда `chown` змінює власника файлу: задає користувача, який володіє файлом. Наприклад, щоб встановити, що користувач `user` володіє файлом `owned-file`, використовують таку команду:

```
chown user owned-file
```

Аналогічно до `chown`, команда `chgrp` змінює групу, якій належить файл. Наприклад, щоб встановити, що група `group` володіє файлом `owned-file`, використовують таку команду:

```
chgrp group owned-file
```

Команда `chmod` змінює «режим» файлу, тобто права доступу до файлу, які надаються користувача. Наприклад, щоб дати групі, якій належить файл, право на запис у файл `owned-file`, команду `chmod` використовують так:

```
chmod g+w owned-file
```

1.3.2. Команди для роботи з файлами і директоріями в операційних системах DOS і Windows

Перегляд змісту

Щоб переглянути зміст файлу в операційних системах DOS і Windows використовують команду `type`. Наприклад, щоб переглянути повний зміст файлу `file`, команду використовують так:

```
type file
```

Щоб переглянути зміст директорії, використовують команду `dir`:

```
dir
```

Таке використання виводить подробиці про кожен елемент у директорії. Однак, часто необхідно переглянути лише імена елементів. Для цього використовують параметр `/b`:

```
dir /b
```

Створення

В операційних системах DOS і Windows немає окремих команд, призначених для створення файлів. Однак, створити файли можна за допомогою перенаправлення потоків виведення. Наприклад, щоб створити пустий файл `new-file`, можна використати команду `type`, яка виводить зміст файлу `NUL` (пустий файл, аналог `/dev/null`) і перенаправляє його у файл `new-file`:

```
type NUL >> new-file
```

Щоб створити директорію, використовують команду `md`:

```
md new-dir
```

Якщо команда `md` запущена у командному рядку, в якому увімкнені розширення командного рядка, вона створюватиме усі проміжкові директорії, якщо вони не існують. У командному рядку Windows версій 2000 і пізніше розширення увімкнені за замовчуванням.

Копіювання

Щоб скопіювати файл, використовують команду `copy`. Наприклад, щоб скопіювати файл `new-file` у файл `copied-file`:

```
copy new-file copied-file
```

Щоб скопіювати директорію, використовують команду `xcopy`. Наприклад, щоб скопіювати директорію `new-dir` у `copied-dir`:

```
xcopy new-dir copied-dir
```

Команда `xcopy` дозволяє копіювати як директорії, так і файли.

Видалення

Щоб видалити файл, використовують команду `del`. Наприклад, щоб видалити файл `new-file`:

```
del new-file
```

Також можна видаляти файли в інтерактивному режимі за допомогою параметра `/P`. Щоб видалити директорію, використовують команду `rd`:

```
rd new-dir
```

Зміна параметрів

Щоб змінити параметри файлів або директорій, використовують команду `attrib`. Наприклад, щоб встановити (+) параметр «тільки для читання» (R) файлу `file1` виконують таку команду:

```
attrib +R file1
```