

Laboratorium 5 - Wejścia i wyjścia procesu

1. Teoria

1.1. `wc`

Zlicza liczbę linii, słów i bajtów (znaków).

Argument	Opis
<code>-l</code>	Zlicza tylko liczbę linii.

1.2. `sort`

Sortuje linie pliku.

Argument	Opis
<code>-n</code>	Sortuje linie traktując je jako liczby
<code>-r</code>	Sortuje w odwrotnej kolejności

1.3. `uniq`

Usuwa powtarzające się po sobie linie.

1.4. `cut`

Wycina fragmenty każdej linii.

Argument	Opis
<code>-f</code> numery pól	Wycina wybrane pola.
<code>-d</code> separator	Dzieli linię na pola używając podanego separatora.

1.5. `wget`

Pobiera zasób.

Argument	Opis
<code>adres</code>	Pobiera zasób znajdujący się pod danym adresem.

1.6. head

Wypisuje pierwsze 10 linii.

Argument	Opis
<code>-n liczba</code>	Wypisuje podaną liczbę pierwszych linii.

1.7. tail

Wypisuje ostatnie 10 linii.

Argument	Opis
<code>-n liczba</code>	Wypisuje podaną liczbę ostatnich linii.

1.8. date

Wypisuje aktualną datę.

Argument	Opis
<code>+FORMAT</code>	Wypisuje datę w określonym formacie (szczegóły w podręczniku).

1.9. /dev/null

Wirtualne urządzenie, które zapomina wszystko to co do niego przekierujemy

2. Praktyka

2.1. Zadanie



`wc`

Spróbuj policzyć liczbę plików i katalogów w twoim katalogu domowym za pomocą polecenia `ls` i `wc`.

2.2. Zadanie

Policz ile użytkowników znajduje się w systemie Linux.

2.3. Zadanie



`cut`

Za pomocą komendy `cut` spróbuj obciąć ze strumienia listy użytkowników same loginy użytkowników.

2.4. Zadanie

Wykonaj to samo co w powyższym punkcie, ale wytnij ścieżkę standardowego shella użytkowników. (wskazówka, polecenie `cut` posiada dwie przydatne opcje `-f` `-d`)

2.5. Zadanie



`sort, uniq`

Policz ile różnych shelli używają użytkownicy w systemie Linux na Twoim komputerze.

2.6. Zadanie

Zapisz listę plików i katalogów znajdujących się w katalogu `/etc` do pliku `lista_etc`.

2.7. Zadanie

Użyj tego samego polecenia jak w punkcie powyżej ale zamiast `/etc` użyj nieistniejącego katalogu. Czy komunikat błędu zapisał się do pliku czy wyświetlił na ekranie?

2.7. Zadanie

Zmodyfikuj polecenie z powyższego punktu tak aby komunikat błędu także zapisał się do pliku.

2.9. Zadanie



`date`

Napisz polecenie, które po każdym uruchomieniu dopisze obecną datę i godzinę do pliku o nazwie `uruchomienia`.