

# Laboratorium 2 - Podstawowe polecenia Linux (obsługa plików i katalogów)

## 1. Teoria

### 1.1. Znak zachęty

---

Znak zachęty to komunikat mówiący o gotowości do przyjęcia polecenia od użytkownika, zazwyczaj jest to znak dolara \$.

Często jest on bardziej rozbudowany (`user@local:~/public_html/$`):

- nazwa użytkownika
- nazwa hosta
- aktualna ścieżka

### 1.2. Automatyczne uzupełnianie - Tab

---

Klawisz `Tab` odpowiada za automatyczne uzupełnianie poleceń i ścieżek. Po wpisaniu początkowych znaków polecenia naciskamy `Tab`. Jeśli istnieje tylko jedno możliwe zakończenie polecenia to jest ono uzupełniane. W przeciwnym wypadku naciskamy klawisz `Tab` ponownie i zostaje wypisana lista możliwych zakończeń polecenia.

### 1.3. Historia

---

Za pomocą klawiszów strzałka w górę i w dół możemy poruszać się po wydanych poleceniach.

### 1.4. Ścieżki

---

Ścieżki do plików lub katalogów mogą być:

- bezwzględne - zaczynają się od korzenia /, np. `/home/epi/00_student`
- względne (względem aktualnego katalogu), np. `podkatalog/plik`
  - `.` - oznacza aktualny katalog
  - `..` - oznacza katalog wyżej, np. `../../pracow/user`

Znak tylda `~` jest automatycznie rozwijany do ścieżki katalogu domowego, np. `~/public_html` jest równoznaczny z `/home/pracow/user/public_html` (gdy katalogiem domowym użytkownika jest `/home/pracow/user`).

## 1.5. Wildcardy

*	Zastępuje dowolny ciąg znaków
?	Zastępuje jeden znak

## 1.6. `man`

Wyświetla podręcznik za pomocą programu `less`.

Argument	Opis
polecenie	dotyczy podanego polecenia

## 1.7. `ls`

Wypisuje zawartość katalogu.

Argument	Opis
ścieżka	odnosi się do podanej ścieżki, domyślnie aktualny katalog
<code>-a, --all</code>	wypisuje również pliki ukryte (zaczynające się od kropki)
<code>-l</code>	wypisuje dodatkowe informacje o plikach, np. prawa dostępu, właściciela, rozmiar, datę modyfikacji
<code>-R, --recursive</code>	wypisuje podkatalogi rekurencyjnie
<code>-h, --human-readable</code>	w połączeniu z argumentem <code>-l</code> rozmiar podawany jest w "ludzkich" jednostkach

## 1.8. `cd`

Zmienia katalog.

Argument	Opis
brak	przechodzi do katalogu domowego użytkownika
ścieżka	przechodzi do podanego katalogu
-	przechodzi do poprzedniego katalogu

## 1.9. `mkdir`

Tworzy katalog.

Argument	Opis
ścieżka	tworzy katalog o podanej nazwie
<code>-p</code>	tworzy katalogi rodziców

## 1.10. echo

---

Wypisuje podany napis (na standardowe wyjście).

Argument	Opis
napis	
-n	nie dodaje na końcu znaku nowej linii
-e	interpretuje znaki specjalne
-E	nie interpretuje znaków specjalnych (domyślne)

### 1.10.1. Znaki specjalne

Ciąg	Opis
\\	odwrócony ukośnik \
\n	nowa linia
\r	powrót karetki
\t	tabulacja

## 1.11. cat

---

Wypisuje zawartość pliku (na standardowe wyjście).

Argument	Opis
brak	wypisuje dane podane na standardowym wejściu, <b>Ctrl+D</b> (EOF - end of file) kończy wprowadzanie danych z klawiatury
ścieżki	wypisuje pliki o kolejnych ścieżkach

### 1.11.1. Przekierowania

polecenie > ścieżka	Zapisuje standardowe wyjście polecenia do pliku pod wskazaną ścieżką
polecenie >> ścieżka	Dopisuje standardowe wyjście polecenia do pliku pod wskazaną ścieżką
polecenie1   polecenie2	Przekazuje standardowe wyjście polecenia 1 na standardowe wejście polecenia 2
polecenie < ścieżka	Przekazuje zawartość pliku pod wskazaną ścieżką na standardowe wejście polecenia

---

### 1.12. `rm`

---

Usuwa plik/katalog.

Argument	Opis
ścieżka	usuwa podany plik/katalog
<code>-r</code>	argument konieczny do usunięcia katalogu z zawartością
<code>-i</code>	generuje pytanie o usunięcie każdego pliku
<code>-f</code>	nie pyta o potwierdzenie usunięcia, dotyczy plików ukrytych

### 1.13. `cp`

---

Kopiuje plik/katalog.

Argument	Opis
ścieżka_skąd ścieżka_dokąd	kopiuje plik z jednej lokalizacji do drugiej
<code>-r</code>	argument konieczny do skopiowania katalogu

### 1.14. `mv`

---

Przenosi plik/katalog.

Argument	Opis
ścieżka_skąd ścieżka_dokąd	przenosi plik z jednej lokalizacji do drugiej

### 1.15. `more`

---

Wyświetla zawartość pliku używając paginacji. Zaleca się używanie programu `less`: "Users should realize that `less(1)` provides `more(1)` emulation plus extensive enhancements."

Argument	Opis
ścieżka	wyświetla zawartość podanego pliku

### 1.16. `less`

---

Umożliwia przeglądanie zawartości plików w konsoli.

Argument	Opis
ścieżka	wyświetla zawartość podanego pliku

Wyszukiwanie odbywa się przez wciśnięcie ukośnika `/` i podanie frazy. Klawisz `N` odpowiada za wyszukanie następnego wystąpienia a kombinacja `Shift+N` za poprzedniego. Z programu wychodzimy naciskając `Q`.

---

### 1.17. `find`

---

Wyszukuje pliki.

Argument	Opis
ścieżka <code>-name</code> wzorzec	wyszukuje pliki w podanym katalogu o podanym wzorcu

### 1.18. `ln`

---

Tworzy dowiązanie (skrót) do pliku/katalogu.

Argument	Opis
ścieżka_celu nazwa	tworzy dowiązanie do celu o podanej nazwie
<code>-s</code>	tworzy dowiązanie symboliczne

### 1.19. `pwd`

---

Wypisuje bezwzględną ścieżkę aktualnego katalogu.

### 1.20. `dirname`

---

Wypisuje ścieżkę do katalogu na podstawie podanej ścieżki (nie operuje na dysku).

Argument	Opis
ścieżka	analizuje podany ciąg znaków

### 1.21. `basename`

---

Wypisuje nazwę pliku na podstawie podanej ścieżki (nie operuje na dysku).

Argument	Opis
ścieżka	analizuje podany ciąg znaków
ścieżka rozszerzenie	analizuje podany ciąg znaków oraz usuwa rozszerzenie

### 1.22. `touch`

---

Tworzy pusty plik.

Argument	Opis
ścieżka	tworzy pusty plik pod podaną ścieżką

## 2. Praktyka

---



Proszę czytać wszystkie komunikaty wyświetlane w konsoli.

### 2.1. Zadanie 0

---



Tab

W ilu najmniej naciśnięciach klawiszy da się wydać polecenie `mkdir /etc/passwd` (tak, nie ma ono sensu)?

### 2.2. Zadanie 1

---



`ls, cd, mkdir`

W katalogu domowym utwórz następującą strukturę katalogów:

- `~/dokumenty`
  - `praca/`
    - `raporty/`
  - `dom/`
    - `zakupy/`
  - `studia/`
    - `podania/`

### 2.3. Zadanie 2

---

Spróbuj wykonać powyższe zadanie używając tylko jednego polecenia zamiast sekwencji `mkdir, ls, cd`.

### 2.4. Zadanie 3

---



`echo`

W katalogu `zakupy/` utwórz plik o nazwie `lista`. Zrób to za pomocą komendy: `echo mleko > lista`

### 2.5. Zadanie 4

---



`cat`

Wyświetl zawartość pliku `lista`.

### 2.6. Zadanie 5

---

Zaobserwuj co się stanie jeśli wykonasz następnie polecenie `echo chleb > lista`

## 2.7. Zadanie 6

---



rm

Usuń plik lista i przetestuj czym różni się poprzednia komenda od użycia komend:

```
echo mleko >> lista
echo chleb >> lista
```

## 2.8. Zadanie 7

---

Usuń katalog dom/ wraz z podkatalogami i plikami. Spróbuj wykonać to także tylko jednym poleceniem.

## 2.9. Zadanie 8

---



cp, mv

Do katalogu raporty/ skopiuj plik /etc/passwd (który zawiera listę użytkowników w systemie). Plik w ścieżce docelowej powinien nazywać się uzytkownicy. Spróbuj skopiować plik jednocześnie nadając mu nową nazwę (jedna komenda).

## 2.10. Zadanie 9

---



more, less

Zmniejsz okno terminala i spróbuj wyświetlić zawartość pliku uzytkownicy. Zawartość nie zmieściła się w jednym oknie. Spróbuj wykonać polecenia:

```
cat uzytkownicy | more  #(bardziej poprawne to more uzytkownicy)
cat uzytkownicy | less  #(bardziej poprawne to less uzytkownicy)
```

## 2.11. Zadanie 10

---



find

Spróbuj wyszukać za pomocą polecenia find wszystkie pliki w Twoim katalogu domowym zaczynające się na literę d. W tym celu użyj wildcard \*

## 2.12. Zadanie 11

---



`ln`

W katalogu `studia/` utwórz katalog `programy/` w którym utwórz katalogi `program-1.0`, `program-1.1`, `program-1.3`. Zrób link symboliczny o nazwie `program` do katalogu z najwyższą wersją (1.3). Spróbuj dodać plik do katalogu `program-1.3` i przekonaj się, że plik ten znajduje się także w katalogu `program`.

## 2.13. Zadanie 12

---

Sprawdź poleceniem `ls -al`, że plik `program` jest rzeczywiście linkiem symbolicznym i wskazuje na katalog `program-1.3`.

## 2.14. Zadanie 13

---



`pwd, dirname, basename`

Na przykładzie pliku `uzytkownicy` zaprezentuj działanie programów `pwd`, `dirname`, `basename`. Odczytaj pełną ścieżkę za pomocą `pwd`, a następnie podaj ją jako argument do dwóch pozostałych programów.

## 2.15. Zadanie 14

---

Na końcu usuń cały katalog `dokumenty/`.

## 2.16. Zadanie 15

---

Czym różni się link twardy (hard link) od symbolicznego?

## 2.17. Zadanie 16

---

Sprawdź do czego służą polecenia: `bc`, `cal`, `df`, `diff`, `du`, `finger`, `grep`, `host`, `htop`, `id`, `ifconfig`, `kill`, `lynx`, `md5sum`, `nano`, `nslookup`, `pico`, `pine`, `ping`, `rmdir`, `reset`, `scp`, `skill`, `su`, `sudo`, `top`, `uname`, `unzip`, `unrar`, `vi`, `vim`, `wc`, `wget`, `who`, `whoami`, `whois`, `write`, `yes`, `zip`