

[Powrót \(Systemy\\_operacyjne\).](#)

# Prawa dostępu, standardowe strumienie, przekierowania, potoki

## ćwiczenia laboratoryjne

### Ważne komendy

- `chmod` - zmienia prawa dostępu do plików oraz katalogów.
- `chown` - zmienia właściciela oraz grupę pliku.
- `tee` - pobiera dane ze strumienia wejściowego i tworzy dwa strumienie wyjściowe: jeden podłączony do standardowego wyjścia, a drugi do wskazanego pliku.
- `tr` - zamienia lub usuwa znaki ze strumienia wejścia.
- `umask` - umożliwia zmianę maski domyślnych uprawnień.

Do wyświetlenia praw dostępu do plików i katalogów można posłużyć się poleceniem: `ls -l`

- typ pliku - pierwszy bit
- prawa dostępu dla właściciela: drugi, trzeci, czwarty bit
- prawa dostępu dla grupy: piąty, szósty, siódmy bit
- prawa dostępu dla pozostałych użytkowników: ósmy, dziewiąty, dziesiąty bit

Typy plików:

znak	znaczenie
-	zwykły plik
d	katalog
c	plik specjalny (urządzenia znakowe)
b	plik specjalny (urządzenia blokowe)
l	dowiązanie symboliczne
s	gniazdo
p	potok nazwany

Prawa dostępu:

wartość		prawo dostępu
literowa	ósemkowa	
r	4	odczyt
w	2	zapis
x	1	wykonanie

Wszystkie kombinacje praw dostępu:

wartość		prawo dostępu
ósemkowa	literowa	
0	---	brak
1	--x	wykonanie
2	-w-	zapis
3	-wx	zapis, wykonanie
4	r--	odczyt
5	r-x	odczyt, wykonanie
6	rw-	odczyt, zapis
7	rwX	odczyt, zapis, wykonanie

Ustawianie praw dostępu:

znak	znaczenie
+	dodawanie uprawnień
-	odbieranie uprawnień
=	przypisanie uprawnień
u	użytkownik (właściciel)
g	grupa
o	pozostali

a	wszyscy
---	---------

## Standardowe strumienie

- 0 - standardowe wejście (stdin).
- 1 - standardowe wyjście (stdout).
- 2 - standardowe wyjście błędów (stderr).

## Instrukcje przekierowań i potoków

- *polecenie* < *plik* - standardowe wejście (stdin), plik musi istnieć.
- *polecenie* > *plik* - standardowe wyjście (stdout), nadpisanie zawartości do pliku.
- *polecenie* >> *plik* - standardowe wyjście (stdout), dopisanie zawartości do pliku.
- *polecenie* 2> *plik* - standardowe wyjście błędów (stderr), nadpisanie zawartości do pliku.
- *polecenie* 2>> *plik* - standardowe wyjście błędów (stderr), dopisanie zawartości do pliku.
- *polecenie1* | *polecenie2* - potok, uruchomienie dwóch poleceń współbieżnie.

## Zadania

**Zadanie 1.** Wypróbuj (dla użytkownika) i wyjaśnij zasadę działania praw dostępu odczytu (r), zapisu (w) i wykonania (x) nadanych do pliku oraz do katalogu. Sprawdź, które z nich powodują, że nie można usunąć pliku/katalogu oraz wyjaśnij różnicę działania praw dostępu.

**Zadanie 2.** Utwórz w swoim katalogu domowym katalog o nazwie *A1*, a następnie w *A1*, katalog o nazwie *A2*. Przejdź do *A2* i odbierz wszystkie prawa dostępu bieżącego katalogu. Następnie (będąc w katalogu *A2*) sprawdź czy możliwa jest modyfikacja praw dostępu oraz przejście do katalogu nadrzędnego *A1*.

**Zadanie 3.** Przywróć katalogowi *A2* wszystkie prawa dostępu dla właścicielowi oraz odbierz wszelkie prawa do katalogu *A1*. Z poziomu swojego katalogu domowego sprawdź czy możliwe jest bezpośrednie przejście do katalogu *A1/A2*.

**Zadanie 4.** Przeczytaj opis polecenia *umask* (komenda *man*). Jakie będą prawa dostępu dla nowo utworzonego pliku oraz katalogu, przy użyciu maski:

- a) 022
- b) 666?

**Zadanie 5.** Wykorzystując odpowiednie komendy i przekierowania zapisz do pliku *katalogi.txt* informacje na temat zawartości Twojego katalogu domowego.

**Zadanie 6.** Wykorzystując przekierowania dopisz do pliku *katalogi.txt* treść:

*zawartość katalogu domowego użytkownika:*

*<twój\_login>* .

Informacje na temat loginu, pobierz wykorzystując odpowiednią komendę.

**Zadanie 7.** Korzystając ze strumienia standardowego wejścia sprawdź ile wierszy, słów i znaków zawiera plik *katalogi.txt*. Wykorzystując przekierowania zapisz te dane do pliku *informacje.txt*. Zawartość pliku: *informacje.txt*, powinna wyglądać następująco:

*Liczba wierszy:*

*<liczba>*.

*Liczba słów:*

*<liczba>*.

*Liczba znaków:*

*<liczba>*.

**Zadanie 8.** Przetestuj polecenia:

```
cat -n katalogi.txt > katalogi2.txt
cat -n katalogi.txt 1> katalogi2.txt
cat -n > katalogi2.txt katalogi.txt
cat -n 1> katalogi2.txt katalogi.txt
cat > katalogi2.txt -n katalogi.txt
cat 1> katalogi2.txt -n katalogi.txt
> katalogi2.txt cat -n katalogi.txt
1> katalogi2.txt cat -n katalogi.txt
```

Wyjaśnij jakie są różnice działania pomiędzy nimi.

**Zadanie 9.** Korzystając z potoku wyświetl zawartość swojego katalogu domowego wraz z informacją odnośnie uprawnień, właściciela, rozmiaru itp., w taki sposób aby można było wyświetlać kolejne wiersze z możliwością przejścia w przód i tył.

**Zadanie 10.** Korzystając z potoku wyświetl:

- a. pierwsze 3,
- b. ostatnie 3

pliki i foldery z katalogu */home*.

zaczynające się na m.

**Zadanie 11.** Wykorzystując przekierowania zapisz do pliku *blad.txt* informację o błędzie wywołanym poprzez wykonanie polecenia `./echo "blad!!!"`

**Zadanie 12.** Wyświetl zawartość dowolnego pliku w taki sposób, aby wszystkie litery były wielkie.

**Zadanie 13.** Korzystając z potoku wyświetl wszystkie pliki i foldery w bieżącym katalogu w jednej linii oddzielone przecinkiem.

Podpowiedź: Znak nowej linii jest oznaczony poprzez: `"\n"`.

**Zadanie 14.** Wyświetl posortowane wiersze z pliku */etc/passwd* przy użyciu:

- a. podstawowego polecenia,
- b. przekierowania strumienia,
- c. potoku.

**Zadanie 15.** Wykonaj (po kolei) poniższe komendy:

```
echo "123456789" > liczby.txt
exec 3<> liczby.txt
read -n 4 <&3
echo -n "abc" >&3
exec 3>&-
```

Zobacz zawartość pliku *liczby.txt*. Następnie wykonaj polecenia przy użyciu deskryptora pliku 4 lub innego wyższego niż 2.

Co można zaobserwować? Wyjaśnij jak działa każde z powyższych poleceń.