

**nano** – Łatwy w obsłudze edytor tekstu.

Użycie:

nano plik.txt

Najczęściej stosowane skróty klawiszowe:

Ctrl-O – Zapis do pliku (jak *Output*)

Ctrl-X – Wyjście. Gdy plik ma niezapisane zmiany, program pyta o zapis.

Ctrl-6 (Ctrl-^) – Rozpoczęcie zaznaczania tekstu.

Ctrl-K – wycina linię lub zaznaczenie

Ctrl-U – wkleja linię lub zaznaczenie. Skopiowanie wygląda więc następująco: Ctrl-K, Ctrl-U, a później wklejanie Ctrl-U.

Ctrl-W – szybkie wyszukiwanie tekstu.

Odpowiednikiem w systemach nie-GNU jest **pico**, wchodzący w skład pakietu pocztowego **pine**.

**mail** – podstawowy program pocztowy

Użycie np.

mail -s "Temat" adres@domena.tld

Otwiera ekran wpisywania wiadomości. Wiadomość wysyłana jest od razu do serwera. Przesłanie wiadomości do odbiorcy następuje po wprowadzeniu linii zawierającej wyłącznie . (kropkę).

Uruchomiony bez parametrów sprawdza, czy nie ma nowych wiadomości dla użytkownika.

W przypadku gdy włączone jest przekazywanie poczty i tak mail się w odebranych nie pojawi.

Gdy będą dostępne nowe wiadomości, mail umożliwi wybranie wiadomości do odczytania. Przewijanie: +/- . Usuwanie: d.

**find** – wyszukiwanie plików

Podstawowym zadaniem programu find jest wyprowadzenie na konsolę listy obiektów w danym w pierwszym parametrze katalogu i rekurencyjnie w głąb. Stąd możemy np. użyć do znalezienia pasującej nazwy plików np. grepa lub wbudowanego w program find narzędzia. W tym celu używamy następujących parametrów;

-name ... – wydrukuje nazwę gdy ta pasuje do wyrażenia podanego po tym parametrze.

-size n - wydrukuje nazwę gdy plik jest rozmiaru n bloków (1 blok – 512 bajtów)

-size nc – jw., lecz w bajtach (np. -size 1048576c), k – kB, M – MB, G – GB.

Niektóre wersje: np. -size +1024c – rozmiar >1024B ; -size -1024c – rozmiar <1024B

-user ... - dopasowanie właściciela pliku

-type f – wyszukiuje konkretne typy obiektów. f – plik, d – katalog, l – dowiązanie itd.

Można użyć dwóch lub więcej wyrażeń łącząc je operatorami -a (AND, i) oraz -o (OR, lub).

Użyteczne parametry:

Parametr -exec uruchamia polecenie dla każdego elementu znalezionego przez find. Używać ostrożnie.

Np.

```
find ~/public_html -type f -name '*.php' -exec chmod a+x {} \;
```

Parametr -prune wywoła wyświetlanie nazwy (lub przetworzenie -exec) pasującego katalogu, ale jego zawartość nie będzie przetwarzana.

Po parametrze -exec argument {} jest zamieniany na wynik programu find. Linia poleceń po -exec musi być zakończona znakami \; . Proszę pamiętać, że jeżeli chcemy coś z wynikiem polecenia "-exec" zrobić, to wynik jest dostępny już po \; - czyli np.:

```
find ./ -name '*.txt' -exec cat {} \; | grep "From"
```

Program find powinien być w naszej konfiguracji wywoływany **zawsze z interesującym nas położeniem** (np. find ./). W nowszych wersjach program zakłada użycie bieżącego katalogu gdy nie podano ścieżki.

**wc** – jak *word count* – liczy słowa, linie lub bajty w pliku lub strumieniu.

**wc -c** plik.txt – wyświetla liczbę bajtów

**wc -m** plik.wxt – wyświetli liczbę znaków

**wc -l** plik.txt – liczbę linii

**wc -w** plik.txt – liczbę słów (oddzielanych spacją lub znakiem nowej linii)

**wc -L** plik.txt – wyświetli długość najdłuższej linii w pliku.

Jeżeli żaden argument nie jest podany, wyświetli odpowiednio ilość linii, słów i znaków. Domyślnie drukowana linia kończona jest nazwą pliku.

**UWAGA!** W starszych wersjach programu **wc**: **-c** liczy znaki (characters), a **-m** nie występuje. Stąd **nie jest bezpiecznie polegać na wc przy szacowaniu rozmiaru pliku**.

**paste** – "sklejanie" tekstu z kilku plików w kolumny. Umożliwia łączenie tekstu z kilku plików w kolumny. Przykładowo, mamy następujące pliki:

**plik A.txt**

Adam

Juliusz

Jerzy

**plik B.txt**

Mickiewicz

Słowacki

Grebosz

operacja **paste -d ' ' \*.txt** (gdy w katalogu jest tylko A.txt i B.txt) lub

**paste -d ' ' A.txt B.txt**

spowoduje wyprowadzenie na konsolę wyniku:

Adam Mickiewicz

Juliusz Słowacki

Jerzy Grebosz

Przy czym argument po **d** **musi być jednym znakiem w ' '.** - z dłuższych zostanie użyty pierwszy znak.

Drugim argumentem jest **-s**, który umożliwia połączenie linijek każdego pliku w jedną i wyprowadzanie jeden plik – jedna linijka (**paste -s A.txt B.txt**):

Adam Juliusz Jerzy

Mickiewicz Słowacki Grebosz

**tar** – jak *Tape Archiver* – Tworzenie i rozpakowywanie plików spakowanych – lecz **nie skompresowanych**.

Parametry: Pierwsza litera:

**-c** – Create – tworzenie archiwum

**-x** – eXtract – rozpakowanie archiwum

**-t** – Type – wyświetla listę plików w archiwum

**-u** – Update – dodaje do archiwum tylko nowsze z podanych plików (lub te, których nie ma).

Na końcu zaś zazwyczaj:

**-f** – File – po tym parametrze podajemy nazwę pliku .tar

Zawsze można dołączyć parametr **-v** wyświetlający zawartość w postaci szczegółowej.

Ostatnim parametrem są nazwy plików/katalogów poddanych spakowaniu / docelowym dla rozpakowania.

Np.

**tar -cf ./paczka.tar plik1.txt plik1.bak ./kopie**

Tworzy paczkę zawierającą pliki plik1.txt, plik1.bak i katalog ./kopie wraz z zawartością.

**tar -xvf ./program.tar**

Rozpakowuje plik do bieżącego katalogu wyświetlając co zostało utworzone.

TAR w systemie SunOS jest połączony z urządzeniami taśmowymi, więc domyślnie czyta z taśmy

(/dev/mt...) i zapisuje na taśmę. Aby tego uniknąć należy nazwę pliku poprzedzić położeniem (./ - bieżący

katalog). TAR powinien przyjmować parametry jako ciąg znaków, zakończony **f** (file). np. **-xvf**, nie **-x -v -f**.

To ograniczenie nie istnieje w GNU TAR.

W systemach GNU/Linux występuje implementacja GNU TAR, posiadająca szereg różnic względem oryginalnego TARA, wśród nich najistotniejsze to:

- Opcja -z automatycznie kompresuje bądź dekompresuje do gzip-a.
- Opcja -j automatycznie kompresująca metodą bzip2.
- Opcja -r pozwalająca dodawać pliki do istniejącego już archiwum.
- Opcja -A pozwalająca dodawać pliki zarchiwizowane do drugiego pliku zarchiwizowanego.

Uwaga: Nieprawidłowość drugiego archiwum nie powoduje błędu programu.

- Niezwiązanie z napędami taśmowymi systemu

W systemie SunOS oryginalny tar nie posiada takiej opcji, jednak w niektórych wersjach istnieje program **gtar** realizujący funkcje wersji GNU.

**Uwaga:** Format klasycznego TARA może nie być zgodny z tym, co gtar generuje z parametrem -u (update) lub w nowszych -a (append – dodaje pliki bez sprawdzania który nowszy).

**gzip/gunzip** – kompresja i dekompresja jednego pliku

Program kompresuje plik podany jako argument, a następnie kasuje go pozostawiając tylko wersję spakowaną o rozszerzeniu .gz. np.

```
gzip plik.png
```

spakuje plik.png do plik.png.gz .

Program gunzip działa w odwrotny sposób. Przyjmuje plik .gz, rozpakowuje go i usuwa.

W niektórych systemach można rozpakować plik opcją gzip -d.

### Wyrażenia regularne

. - kropka oznacza „dopasuj dowolny znak”

[abc] – dopasuj dowolny ze znaków a, b lub c.

[^abc] – dopasuj dowolny znak, ale nie a, b lub c. - np. [^0-9] – dowolne znaki nie będące cyframi.

[0-9] – cyfry

[a-z] – małe litery

[A-Z] – wielkie litery

\ - użycie znaku specjalnego jako znaku do dopasowania, np. [\\] - dopasowuje klamry

Ilość:

\* - występowanie 0 lub więcej znaków.

+ - występowanie 1 lub więcej znaków

? - 0 lub 1 znak.

A to działa w trybie rozszerzonym (grep -E):

{min,max} – ilość znaków w zakresie

{min, } - minimalna ilość znaków.

{,max} – maksymalna ilość znaków

{ilość} – konkretna ilość znaków.

Np. [0-9]{2}-[0-9]{3}- dopasowanie kodu pocztowego

Istnieje znaczna ilość wersji programu grep, charakteryzują się one odmiennieściami wspieranych wyrażeń regularnych. Większość wersji we wspieranych systemach jest przynajmniej w trybie rozszerzonym zgodna w dużej części z wyrażeniami Perl-owymi (GNU Grep). W przypadku grep'a w starszych wersjach POSIX (jaki występuje w SunOSie) **nie jest możliwe działanie tym operatorem** nawet gdy uruchomimy grep z parametrem -E (extended regular expressions).

### Początek i koniec – znaki ^ i \$:

```
grep "^st" tekst.txt
```

dopasuje jak linia zaczyna się "start" i "stop". Ale nie "echo start".

Zaś znak \$ oznacza koniec linii:

```
grep ')$' skrypt.sh
```

wypisze wszystkie linie kończące się znakiem ), co okaże się bardzo użyteczne przy sprawdzaniu później instrukcji typu switch..case w programach shell'owych. Pamiętajmy o apostrofie gdy używamy \$ by zatrzymać ew. rozszerzenie zmiennej - napotkanie np. \$" (zakończenie wyrażenia regularnego \$) spowoduje, że powłoka będzie próbowała znaleźć zmienną o nazwie ". Dla powłoki dowolny tekst po \$ jest nazwą zmiennej, którą można użyć w programach własnych i do automatyzacji.

### Alternatywa

W starszych wersjach programu grep (co pozostało np. w systemach IBM) stosowano {}. Większość dystrybucji umożliwia jednak zastosowanie znaku alternatywy | - np. "a la ma (kota|psa)". Na naszym serwerze **nie działa grep -E**, podobny efekt można jednak uzyskać programem **egrep**. W nowszych systemach polecenie egrep wciąż istnieje, jest ono jednak po prostu tłumaczone jako grep -E.

**Zadanie:** (jeżeli posiadasz plik mbox) Za pomocą programu egrep uzyskaj ze swojej skrzynki mailowej (plik mbox) linie zawierające adresy e-mail. Wybierz dwa i zastosuj alternatywę (**egrep!**), by wyświetlić tylko linie zawierające te dwa adresy e-mail. W przypadku alternatywy musimy użyć "" dla wyrażenia regularnego.

### Publikowanie strony WWW:

**Uwaga: To wymaga udostępnienia możliwości wejścia do katalogu public\_html (parametr +x na katalogu domowym) czyli przejścia przez katalog domowy!**

Utworzyć katalog public\_html.

W nim utworzyć plik index.html. Musi być on do odczytu przez wszystkich.

Adres: [http://sendzimir.metal.agh.edu.pl/~nazwa\\_konta](http://sendzimir.metal.agh.edu.pl/~nazwa_konta)

### Zadania:

Spakuj pliki z zadania 1 do pliku tar.gz

Przenieś paczkę do nowego podkatalogu i tam ją rozpakuj.

Wykonaj to ćwiczenie za pomocą tar'a oraz, drugi raz, gtar'a używając jednego polecenia dla archiwizacji i skompresowania.