

FIND

<http://www.cs.put.poznan.pl/akobusinska/downloads/find.pdf>

1. Znajdź za pomocą programu locate wszystkie pliki które zawierają w nazwie słowo netscape
`locate netscape`
2. Ogranicz wynik polecenia 1 do tych programów, które występują w podkatalogach katalogu /usr,a następnie do tylko tych plików, których nazwa brzmi dokładnie netscape
`locate „usr/*/netscape”`
3. Znajdź trzy literowe programy, znajdujące się w podkatalogach bin katalogu /usr
`locate „usr/bin/???”`
4. Znajdź wszystkie pliki zwykłe w systemie, które mają w nazwie ciąg znaków “conf” i wyświetl je na ekranie.
`find -type f -name „*conf*”`
5. Znajdź wszystkie podkatalogi w katalogu /var.
`find /var -type d`
6. Odszukaj w swoim katalogu domowym wszystkie pliki zwykłe, mające w nazwie ciąg znaków “txt”.
`find ./ -type f -name „*txt*”`
7. Odszukaj w swoim katalogu domowym wszystkie pliki zwykłe, mające w nazwie ciąg znaków “txt”, wyświetl ich nazwy i zawartość.
`find ./ -type f -name „*txt*” -exec cat {} \;`
8. Odszukaj w katalogu /tmp wszystkie podkatalogi i wyświetl ich opis
`find /tmp/* -type d -exec ls {} -dl \;`
9. Odszukaj w systemie wszystkie pliki (i katalogi), które są Twoją własnością, po czym wyświetl ich nazwy.
`find /* -type f -user infXXX;`
10. Odszukaj w katalogu /etc wszystkie pliki, mniejsze niż 7000 bajtów.
`find /etc -type f -size -7k`
11. Znajdź w swoim katalogu domowym (bez podkatalogów) te pliki, które były modyfikowane w ciągu ostatnich dziesięciu dni i wyświetl ich nazwy.
`Find ./ -maxdepth 1 -mtime 10`
12. Znajdź w katalogu bieżącym pliki zwykłe, które były modyfikowane ponad 4 dni temu.
`Find ./ -type f -mtime +4`
13. Znajdź w swoim katalogu domowym wszystkie pliki, które nie były używane w ciągu ostatnich 20 dni.
`Find ./ -type f ! -ctime -20`
14. Znajdź w swoim katalogu domowym wszystkie pliki zwykłe, które były modyfikowane ponad 8 dni temu i wyświetl ich zawartość, pytając wcześniej o potwierdzenie.
`find ./ -type f -maxdepth 1 -mtime +8 -ok cat {} \;`
15. Znajdź w swoim katalogu domowym wszystkie pliki, które były modyfikowane później niż plik /etc/passwd, wyświetl ich nazwy oraz sprawdź, jakiego są typu. (typ pliku można uzyskać stosując polecenie file)

```
find ~ -not -type d -newer /etc/passwd -print -exec file -b '{}' ';' ;'
```

16. Znajdź wszystkie pliki w katalogu /etc, które są większe niż 10 bajtów i mają więcej niż 1 dowiązanie (link). Wyświetl ich opis (z prawami, atrybutami, itp).

```
find /etc -not -type d -size +10c -links +1 -exec ls -l '{}' ';' ;'
```

17. Znajdź w katalogu /etc wszystkie podkatalogi i pliki o nazwach zaczynających się na literę a, nie będące katalogami.

?

18. Znajdź w swoim katalogu domowym (bez podkatalogów) wszystkie pliki zwykłe, mniejsze niż 50 bajtów, które były modyfikowane w ciągu ostatnich 8 dni i wyświetl po 5 pierwszych linii z każdego z nich.

// ?

19. Znajdź w swoim katalogu domowym wszystkie pliki większe niż 50 bajtów, które nie były używane przez ponad 14 dni, albo modyfikowane przez ponad 10 dni i wyświetl ich opis.

// ?

20. Wyszukaj w całym systemie plików te, które posiadają więcej niż 3 dowiązania

```
find / -links +3
```

21. Usuń z dysku wszystkie pliki o nazwach core lub z przyrostkiem .out które nie były używane w ciągu ostatnich siedmiu dni. Przy usuwaniu system powinien zapytać o potwierdzenie

```
find ~ '(' -name core -or -name *.out ')' -exec rm -i '{}' ';' ;'
```

22. Znajdź katalogi nie posiadające praw zapisu dla użytkownika lub grupy (albo obu), i przyznaj te prawa.

// ?

23. Podaj rozmiar katalogów znajdujących się w Twoim katalogu domowym na I poziomie, oraz na końcu listy wyświetl ich wartość sumaryczną.

```
find ~ -type d -maxdepth 1 -mindepth 1 -exec du --summarize --human-readable '{}' ';' ;  
du --summarize --human-readable ~
```

GREP

<http://www.cs.put.poznan.pl/akobusinska/downloads/grep.pdf>

1. Wyświetl linie i numery linii z pliku /etc/passwd, które zawierają napis "jan".

```
grep -n „jan” /etc/passwd
```

2. Policz, ilu użytkowników systemu używa powłoki bash (zgodnie z zapisami w pliku /etc/passwd).

```
grep -c „/bin/bash$” /etc/passwd
```

3. Wyświetl te linie z pliku /etc/group, które nie zawierają słowa "root".

```
grep -v „root” /etc/group
```

4. Wyświetl te linie z pliku /etc/passwd, które dotyczą Twojego konta. Dane w pliku oddzielone są dwukropkami, ale na ekranie należy je zastąpić znakiem #.

```
getent passwd | grep „infXXX” | tr „:” „#”
```

5. Wyprowadź te wiersze pliku plik.txt, które zawierają napisy „Parametr =, lub „parametr =,.

```
grep „[P|p]arametr =” plik.txt
```

6. Wyświetl listę podkatalogów katalogu bieżącego używając grep-a.

```
ls -l | grep „^d”
```

7. Wyświetl te wiersze pliku plik.txt, które kończą się przecinkiem

```
grep „,$” plik.txt
```

8. Znajdź w plikach pl* wiersze rozpoczynające się od cyfry

```
grep „^[0-9]” pl*
```

9. Podaj nazwy plików pl* i numery linii kończących się na k lub K

```
grep „[k|K]$" pl* -n
```

10. Wyszukaj w plikach podkatalogów linie zawierające *

```
grep „\*” -r
```

11. Wyświetl nazwy plików nie zawierających wyrazów plik

```
grep „plik” -v -l
```

12. Znajdź pliki, w których na 9 pozycji występuje litera g

```
grep „^.\{8\}g” *
```

13. Znajdź pliki zaczynające się literami „pl”, w których występuje co najmniej jedna linia zawierająca ciąg znaków „aaa”, a następnie usuń te pliki

```
grep „aaa” pl* -l | xargs rm
```

14. Na podstawie pliku /etc/passwd wypisz nazwy tych użytkowników, którzy mają puste hasło lub zablokowane konto (blokada konta realizowana jest przez wpisanie w miejscu hasła znaku „*”).

```
grep „:\*:" /etc/passwd
```

15. Podaj polecenie pozwalające przeszukać katalog główny i jego podkatalogi, wypisać nazwy wszystkich plików starszych niż 2 tygodnie, a dodatkowo spośród tak wybranych plików dla plików o rozszerzeniu .txt lub .text, wyświetlić ich zawartość. Strumień błędów usunąć.

```
find ./ -mtime +14 \( -name "*.txt" -o -name "*.text" \) -exec cat {} \; 2>&-
```

16. Wyszukaj linie, które mają identyczne wartości w polach 1 i 3 oraz 2 i 4. (niech separatorem pól będzie znak :)

```
awk 'BEGIN {FS=":"}; ($1==$3)&&($2==$4)' file //awk rlz!
```

17. Znajdź wiersze zawierające tylko białe znaki(tabulacje i spacje)

```
grep "^\s\+$" *
```

18. Znajdź wiersze puste

```
grep „^$” *
```

19. Znajdź wiersze zawierające co najmniej jeden znak

```
grep "\S\+" *
```

20. Znajdź wiersze zawierające liczby rzymskie

```
grep "[IVXLCDM]" * //rozwiązanie bardziej wyrafinowane wymagałoby bycia nindzą  
w yaccu i lexie (liczba IIII jest niepoprawna, grepem się tego sprawdzić nie da)
```

21. Znajdź wiersze zawierające liczby całkowite ewentualnie poprzedzone znakiem

```
grep "[+-]\?[:digit:]\+" * //zakładam, że 3.141 to dwie liczby całkowite (?)
```

22. Z pliku .html usuń wszystkie komentarze.

```
cat file | tr '\n' ' ' | sed 's/<!--/\n<!--/g' | sed 's/-->/-->\n/g' | grep -P -v  
"<!--.*/?-->"
```

23. Usunąć komentarze z programu w języku C.

```
// podobnie jak wyżej?
```

PROCESY

http://www.cs.put.poznan.pl/akobusinska/downloads/procesy_zad.pdf

1. Policz ile kopii powłoki bash jest aktualnie uruchomionych (dla wszystkich użytkowników).

```
ps -e | grep „bash$” -c
```

2. Sprawdź, jakie polecenia uruchomił dowolny inny użytkownik, pracujący w tej chwili w systemie.

```
ps au | grep -v “^infXXX” | tr -s ' ' '\t' | cut -f11
```

3. Wyświetl listę użytkowników, którzy mają uruchomiony program vi

```
ps -ef | grep “vi$” | tr -s ' ' '\t' | cut -f1
```

4. Sprawdź, jakie procesy pracują w tle.

```
jobs
```

5. Uruchom w tle wyszukiwanie w całym systemie plików o nazwach pasujących do wzorca “*user*”, błędy przekieruj na urządzenie puste, wyniki do pliku wyniki. Uruchamiając, obniż priorytet tego polecenia do najniższego.

```
nice -n 19 find . -name “*user*” 2> /dev/null > file&
```

6. Podaj nazwy trzech najmniejszych plików w katalogu posortowane wg nazwy

```
ls -l | tr -s ' ' '\t' | grep “^-” | sort -n -k5 | head -3
```

7. Podaj pięciu użytkowników o największej liczbie uruchomionych procesów

```
ps au | tr -s ' ' '\t' | cut -f1 | sort | awk ‘$0 == prev {count++; prev = $0} $0 != prev {print prev “ ” count; count=1; prev=$0}’ | sort -r -n -k2 | head -5
```

8. Wyświetl zawartość 3 największych podkatalogów katalogu bieżącego

```
du --max-depth 1 | sort -n -k1 | tail -4 | head -3 | cut -f2 | xargs ls
```

9. Wyświetl nazwy tych użytkowników, którzy domyślnie używają innego interpretera niż bash

```
cat /etc/passwd | tr ‘:’ '\t' | cut -f1,7 | grep -v “bash$” | cut -f1
```

10. Wyświetl nazwy wszystkich plików nagłwkowych posortowane wykorzystywanych w plikach bieżącego katalogu

```
// ni chuja nie rozumiem polecenia.
```

11. Wyświetl statystykę używanych komend (bez argumentów) w postaci posortowanej listy: ilość komenda (wsk. należy użyć polecenia history)

```
history | tr -s ' ' '\t' | cut -f3 | sort | awk ‘$0 == prev {count++; prev = $0} $0 != prev {print prev “ ” count; count=1; prev=$0}’ | sort -n -r -k2
```

12. W podanym katalogu utwórz podkatalogi wszystkim użytkownikom ze swojego roku i dodatkowo zapisz w pliku o nazwie users.txt posortowaną listę tych użytkowników.

```
sort users.txt | xargs -n 1 mkdir \ \ jeśli tylko użytkownicy są w pliku sers.txt...
```

13. Wyświetl wszystkie pliki, których właścicielem jest dany użytkownik, a pliki znajdują się w katalogu nie należącym do niego

```
// ?
```

14. Sprawdź czy któryś z użytkowników jest załogowany w systemie więcej niż jeden raz. Dla każdego takiego użytkownika należy wyświetlić jego identyfikator i listę terminali na których pracuje.

```
// ?
```