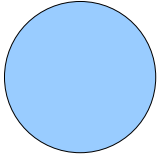
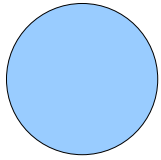


# Polecenia w Linuxie



history



!!

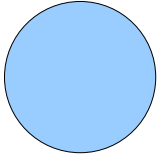
!107

!!:p

echo !!:1

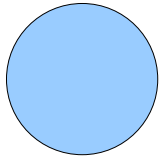
!ac:p

!ac



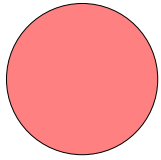
file

file a.txt



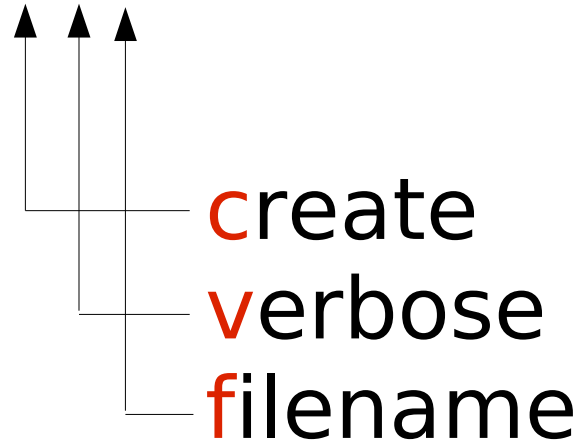
diff

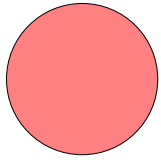
diff a.txt b.txt



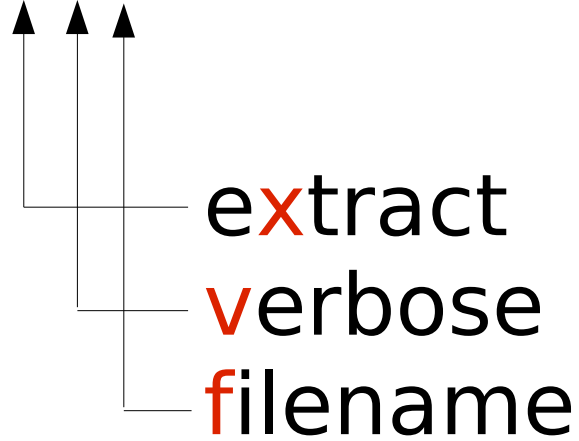
tar                    tape archive

tar cvf a.tar \*.txt

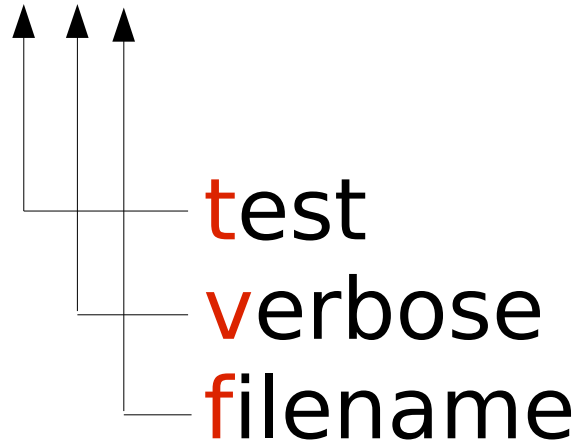


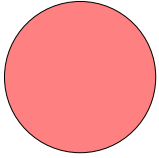


tar xvf a.tar

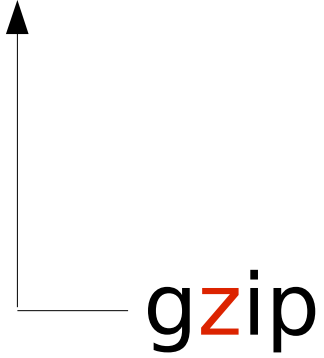


tar tvf a.tar

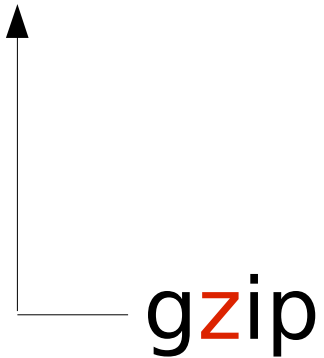




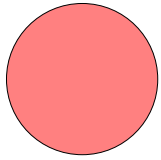
tar czvf a.tar.gz \*.txt



tar xzvf a.tar.gz







gzip

gunzip

.gz

bzip2

bunzip2

.bz2

zip

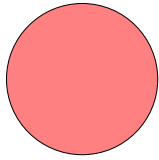
unzip

.zip

compress

uncompress

.Z

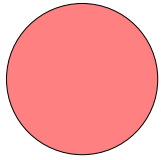


```
tar cvf a.tar *.txt  
gzip a.tar           → a.tar.gz
```

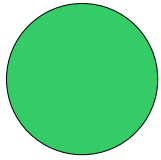
```
= tar czvf a.tar.gz *.txt
```

```
tar cvf a.tar *.txt  
bzip2 a.tar          → a.tar.bz2
```

```
= tar cjvf a.tar.bz2 *.txt
```



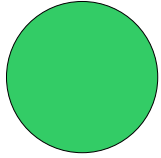
`.tar.gz = .tgz`



uruchamianie procesu w tle

&

gzip a.tar &



# przekierowania

V

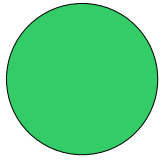
^

V

V

^

^

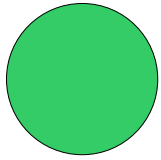


```
cat a.txt > b.txt
```

```
egrep "hello" a.txt > b.txt
```

```
egrep "world" a.txt >> b.txt
```

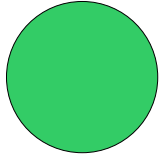
```
cat a.txt > /dev/null
```



```
cat < a.txt
```

```
cat < a.txt > b.txt
```

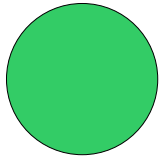
```
cat < a.txt >> b.txt
```



```
cat << EOF
```

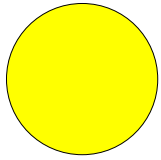
```
cat << EOF > b.txt
```





> nadpisywanie  
< czytanie z pliku

>> dopisywanie  
<< czytanie do znacznika

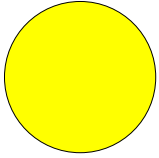


kill

kill 1234

kill -9 1234

killall tcsh

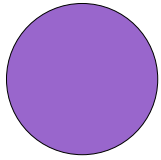


su  
whoami

superuser

su  
su -

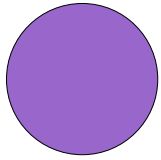
su greg  
su - greg



awk

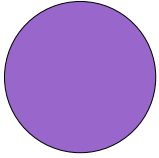
```
cat c.txt | awk '{ print $2 }'
```

```
cat c.txt | awk '{ print $2+10 }'
```



awk

```
cat a.txt | awk -v FS="d" '{ print $2 }'
```



# man

## man awk

MAWK(1)

USER COMMANDS

MAWK(1)

### NAME

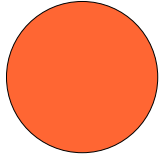
mawk - pattern scanning and text processing language

### SYNOPSIS

```
mawk [-W option] [-F value] [-v var=value] [--] 'program text' [file
...]
mawk [-W option] [-F value] [-v var=value] [-f program-file] [--] [file
...]
```

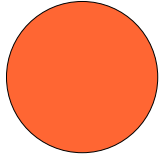
### DESCRIPTION

mawk is an interpreter for the AWK Programming Language. The AWK language is useful for manipulation of data files, text retrieval and processing, and for prototyping and experimenting with algorithms. mawk is a new awk meaning it implements the AWK language as defined in Aho, Kernighan and Weinberger, The AWK Programming Language, Addison-Wesley Publi



```
finger  
ping  
traceroute  
ifconfig  
netstat
```

```
finger greg  
ping astro.as.ap.krakow.pl  
traceroute astro.as.ap.krakow.pl  
ifconfig -a  
netstat -i
```

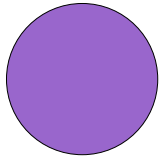


telnet  
ssh

ssh greg@astro.as.ap.krakow.pl

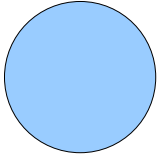
ssh -l greg astro.as.ap.krakow.pl





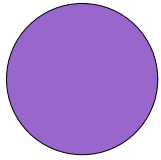
bash

```
while [ 1 ]; do  
  ls -l a.txt  
  sleep 5  
done
```



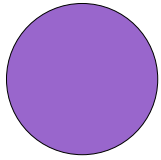
sleep

sleep 5



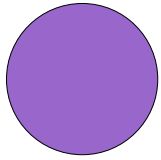
bash

```
for f in `ls *.txt`; do  
    echo $f  
done
```



bash

```
if [ -e a.txt ]; then  
    echo "Hello"  
fi
```



```
#!/bin/bash
```

```
#!/bin/tcsh
```

```
#!/bin/sh
```

```
#!/bin/csh
```

```
chmod a+x
```