Podstawy programowania w języku skryptowym Bash cz.2

Tomasz Dudała, Rafał Wojtyna

Nokia

Wrocław, April 14, 2016

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Operacje na zmiennych tekstowych w Bashu

Tworzenie wycinków

```
1$ dog="woof wooof!"
2$ echo ${dog:3}
3f wooof!
4$ echo ${dog:3:5}
5f woo
```

NOKIA

Operacje na zmiennych tekstowych w Bashu

Obcinanie ciągu

```
1$ dog="woof wooof!"
2$ echo ${dog#*o}
3of wooof!
4$ echo ${dog##*o}
5f!
6$ echo ${dog%%o*}
7w
8$ echo ${dog%o*}
9woof woo
```

NOKIA

Operacje na zmiennych tekstowych w Bashu

Wyszukiwanie i zamiana

```
1$ dog="woof wooof!"
2$ echo ${dog/o/a}
3waof wooof!
4$ echo ${dog//o/a}
5waaf waaaf!
```

NOKIA

Operacje na zmiennych tekstowych w Bashu

Zamiana wzorca na początku lub na końcu ciągu

```
1$ dog="woof wooof!"
2$ echo ${dog/#woof/wooof}
3wooof wooof!
4$ echo ${dog/#"wooof!"/"woof!"}
5woof wooof!
6$ echo ${dog/%"wooof!"/"woof!"}
7woof woof!
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Tablice w Bashu

- Są jednowymiarowe
- Nie trzeba ich deklarować
- Nie mają z góry ustalonych wymiarów
- Istnieją dwie metody iniclalizacji elementów tablicy
- Indeksowane są od ’zera’

NOKIA

Tablice w Bashu - operacje

Inicjalizacja elementów

```
a=(1 2 3)
a[0]=1; a[1]=2; a[2]=3
```

Odwołanie do elementów

```
echo ${a[0]}      # Pierwszy element tablicy
echo $a           # j.w.
echo ${a[-2]}     # Przedostatni element
echo ${#a[0]}     # Dlugosc zerowego elementu
echo ${#a[*]}     # Liczba elementów
echo ${a[*]}      # Wszystkie elementy
# Można też inaczej:
for i in ${a[@]}; do echo $i; done
```

Usuwanie elementów tablicy

```
unset a[1]
unset a[*]
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Wyrażenia regularne

Definicja:

Wyrażenia regularne (ang. regular expressions, w skrócie regex lub regexp) – wzorce, które opisują łańcuchy symboli. Teoria wyrażeń regularnych jest związana z teorią języków regularnych. Wyrażenia regularne mogą określać zbiór pasujących łańcuchów, mogą również wyszczególniać istotne części łańcucha.

`https://pl.wikipedia.org/wiki/Wyrażenie_regularne`

NOKIA

Wyrażenia regularne

Typy wyrażeń regularnych (komenda grep):

- basic (BRE)
- extended (ERE) - w implementacji GNU ta sama funkcjonalność co BRE
- Perl regular expressions (PCRE)

Dokumentacja: man pcrepattern

Wikipedia:

Because of its expressive power and (relative) ease of reading, many other utilities and programming languages have adopted syntax similar to Perl's — for example, Java, JavaScript, Python, Ruby, Microsoft's .NET Framework, and XML Schema.

NOKIA

Wyrażenia regularne

PCRE:

```
\d      any decimal digit
\D      any character that is not a decimal digit
\s      any white space character
\S      any character that is not a white space charact
\w      any "word" character
\W      any "non-word" character
^       assert start of string
$       assert end of string
.       match any character except newline
?       0 or 1 quantifier
*       0 or more quantifier
+       1 or more quantifier
{       start min/max quantifier
```

NOKIA

Wyrażenia regularne

Zwykły ciąg znaków to też wyrażenie

```
1$ [[ "foo,.bar" =~ foo ]] ; echo $?
20
```

Własne klasy znaków. Definiowanie ilości wystąpień

```
1$ [[ "foo,.bar" =~ [\.]{2,3} ]] ; echo $?
21
3$ [[ "foo,.bar" =~ [\.,]{2,3} ]] ; echo $?
40
```

Zakotwiczenie wzorca.

```
1$ [[ "foo,.bar123" =~ [0-9]{1,}$ ]] ; echo $?
20
3$ [[ "foo,.bar123" =~ ^[0-9]{1,} ]] ; echo $?
41
```

NOKIA

Wyrażenia regularne

Znak plus i znak gwiazdki

```
1$ [[ "foo,.bar123" =~ [0-9]+$ ]] ; echo $?
20
3$ [[ "foo,.bar123" =~ ^[0-9]* ]] ; echo $?
40
```

Dopasowanie rozszerzenia pliku

```
1$ [[ "foo,.bar123" =~ \.jpg$ ]] ; echo $?
21
3$ [[ "foo,.bar123.jpg" =~ \.jpg$ ]] ; echo $?
40
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Programy powłoki

sed (stream editor)

- służy do edycji i transformacji tekstu
- działa wiersz po wierszu
- standardowo operacje wykonuje "w powietrzu", źródło nie ulega modyfikacji
- składnia: sed [parametr] wzorzec plik

```
1 sed '1d' plik                    # Usuwa pierwszą linię pliku
2 sed '/Tomek/d' plik              # Usuwa linie zawierające "Tomek"
3 sed 's/Tomek/Romek/' plik        # Zamienia "Tomek" na "Romek"
4 echo "tomek_123" | sed -r 's_[0-9]+_liczba_'
5 # Zamienia ciag liczbowy na "liczba", można użyc innych
    separatorów
6 echo "jeden dwa trzy" | sed -r 's/([a-zA-Z]+).*/\1/'
7 # Zostawiamy tylko pierwszy wyraz
```

NOKIA

Programy powłoki

grep (global regular expression print)

- służy do wyszukiwania w tekście linii zawierających ciągi znaków
- działa wiersz po wierszu
- składnia: grep [parametr] wzorzec plik

```
1 grep 'Tomek' plik
2 # Wyswietli linie zawierające "Tomek"
3 grep '.omek' plik
4 # Wyswietli linie w których pierwszy znak jest dowolny(oprocz
      nowej linii), a dalej "omek", np "Tomek", "Romek"
5 grep 'main' -n -i -r .
6 # wyswietlamy wystapienia "main" "case-insensitive" z nr. linii
      szukajac plikow rekursywnie w bieżącym katalogu
7 cat plik | grep 'main' -A 1
8 # Linia z "main" i jedna linia po
9 grep 'tekst' *
```

Programy powłoki

find (search for files in a directory hierarchy)

- służy do wyszukiwania plików w drzewie katalogów
- pozwala wykonać operacje na wynikach wyszukiwania
- składnia: find [parameters] [path]

```
1 find                          # pliki w bieżącym katalogu i ponizej
2 find dir -name "*.sh"         # pliki konczace sie na .sh w kat. dir
3 find -type d -maxdepth 2 # katalogi na maksymalnie 2 poziomie
4 find . -exec file "{}" \;
```

NOKIA

Programy powłoki

watch (execute program periodically)

- cyklicznie uruchamia komendę i wyświetla wynik na pełnym ekranie
- użyteczny do śledzenia wszelkiego rodzaju zmian

Uruchom listowanie pliku "plik" co 20 sec

```
1watch -n20 "ls plik"
```

Monitoruj zmiane wielkości pliku co 10 sec

```
1watch -n10 --difference=cumulative "du -sb plik"
```

NOKIA

Programy powłoki

cut (remove sections from each line of files)

- służy do ekstrakcji tekstu z linii

Wytnij z linii znaki od 3 do 5 i wypisz

```
1 cut -c 3-5 plik
```

Potnij na pola korzystajac z ":" i wyswietl drugie pole

```
1 echo "user:group:homedir" | cut -d":" -f2
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Wyrażenia regularne w programach powłoki

Podstawowe narzędzia

Dopasowanie do wzorca:

```
1$ echo "foo,.bar123.jpg" | grep -P '\W{2}'
2foo,.bar123.jpg
```

Proste transformacje:

```
1$ echo "foo,.bar123.jpg" | tr [:lower:] [:upper:]
2FOO,.BAR123.JPG
```

Znajdź i zamień:

```
1$ echo "foo,.bar123.jpg" | sed -r 's|[,\.]|#|g'
2foo##bar123#jpg
```

NOKIA

Wyrażenia regularne w programach powłoki

Zgodność z PCRE

```
1$ perl -e '"foo bar 123" =~ /\d+/ or die' &>/dev/null ; echo $?
20
3$ perl -e '"foo bar " =~ /\d+/ or die' &>/dev/null ; echo $?
4255
```

```
1$ [[ "foo bar 123" =~ \d+ ]] ; echo $?
21
3$ [[ "foo bar 123" =~ [0-9]+ ]] ; echo $?
40
```

```
1$ echo "foo bar 123" | grep -E "\d+"
2$ echo "foo bar 123" | grep -E "[0-9]+"
3foo bar 123
4$ echo "foo bar 123" | grep -P "\d+"
5foo bar 123
```

NOKIA

Wyrażenia regularne w programach powłoki

Lazy VS greedy

```
$ perl -e 'print $1."\n" if "foooo bar 123" =~ /(\w+)/'
foooo
$ perl -e 'print $1."\n" if "foooo bar 123" =~ /(\w+?)/'
f
```

```
$ url="http://www.suepearson.co.uk/product/174/71/3816/"
$ echo "$url" | sed -r 's|(http://.*/).*|\1|'
http://www.suepearson.co.uk/product/174/71/3816/
$ echo "$url" | perl -pe 's|(http://.*?/).*|\1|'
http://www.suepearson.co.uk/
```

NOKIA

Wyrażenia regularne w programach powłoki

Grupowanie i alternatywa

```
$ echo "foo bar" | grep -P 'foo\s(foo|bar)'
foo bar
$ echo "foo foo" | grep -P 'foo\s(foo|bar)'
foo foo
$ echo "foo foa" | grep -P 'foo\s(foo|bar)'
$
```

```
$ echo "wof wooof!" | sed -nr 's#^.*(woof|wooof).*$#\1#p'
wooof
```

NOKIA

Wyrażenia regularne w programach powłoki

Dopasowanie negatywne

```
1$ echo "foo bar" | perl -ne 'print if /^(?!foo)/'
2$ echo "bar" | perl -ne 'print if /^(?!foo)/'
3bar
```

```
1$ echo "foo bar" | grep -P '^(?!foo)'
2$ echo "bar" | grep -P '^(?!foo)'
3bar
```

```
1$ echo "bar" | grep -vP '^foo'
2bar
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Programy jednolinijkowe

Wykorzystanie wyników działania innych poleceń

```
$ last | grep reboot | head -n 1
reboot   system boot  3.19.0-21-generi Wed Apr 13 09:28 - 13:41
      (04:13)
```

```
$ du -shl /home/$(whoami)/.config
66M /home/rwojtyna/.config
```

```
$ echo "foo bar" | sed -r 's/(foo|bar)/'"$(hostname)"'/g'
vinson vinson
```

NOKIA

Programy jednolinijkowe

Wykorzystanie polecenia xargs

```
1$ echo "grep" | xargs whatis
2grep (1)               - print lines matching a pattern
3$ echo "re" | xargs -I what whatis gwhatp
4grep (1)               - print lines matching a pattern
5$ whoami | xargs -I who du -shl /home/who/.config
666M /home/rwojtyna/.config
```

NOKIA

Programy jednolinijkowe

Warunki logiczne i pętle

```
1$ [ "$STY" ] || echo "To nie screen"
2To nie screen
3$ [ ! "$STY" ] && echo "To nie screen"
4To nie screen
```

```
1$ ls foo &>/dev/null || echo "Brak foo"
2Brak foo
3$ ls foo &>/dev/null || ( echo "Brak foo"; echo "i co teraz?" )
4Brak foo
5i co teraz?
```

```
1$ while read line; do echo "$line"; done <"date.txt"
2917244:5392:4019
3917244:5392:4019
4917244:5392:4019
5917244:5392:4019
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Programy powłoki

Here document - blok kodu specjalnego przeznaczenia

- używa składni przekierowania strumienia
- służy do zasilenia komendy, interaktywnego programu listą komend

```
1 cat <<TUTAJ > plik
2 echo pierwsza linia
3 echo druga linia
4 TUTAJ
```

```
1 tr a-z A-Z <<LIMIT
2 osiem siedem szesc
3 aaa bbb cwd
4 LIMIT
```

NOKIA

Agenda

1. Działania na ciągach znakowych
2. Tablice Bashowe
3. Wyrażenia regularne, wykorzystanie w Bashu
4. Przykładowe programy powłoki
5. Wyrażenia regularne w programach powłoki
6. Programy jednolinijkowe
7. HEREDOC
8. Obsługa we/wy

NOKIA

Obsługa wejścia i wyjścia

Przekierowania wyjść

```
$ perl -e 'print "Hello! \n" ;die'
Hello!
Died at -e line 1.
$ perl -e 'print "Hello! \n" ;die' 2>/dev/null
Hello!
$ perl -e 'print "Hello! \n" ;die' 1>/dev/null
Died at -e line 1.
```

```
$ perl -e 'print "Hello! \n" ;die' > tmp
Died at -e line 1.
$ cat tmp
Hello!
$ perl -e 'print "Hello! \n" ;die' &> tmp
$ cat tmp
Died at -e line 1.
Hello!
```

NOKIA

Obsługa wejśćia i wyjścia

Przekierowania wyjść

```
$ perl -e 'print "Hello! \n" ;die' | tee tmp
Died at -e line 1.
Hello!
$ cat tmp
Hello!
```

```
$ perl -e 'print "Hello! \n" ;die' 2>&1 | tee tmp
Died at -e line 1.
Hello!
$ cat tmp
Died at -e line 1.
Hello!
```

NOKIA

Literatura

- https://www.dropbox.com/sh/mxeg2na4jmklhmr/AADh687rYXD8jaYoVtLkamvOa?dl=0
- http://tldp.org/LDP/Bash-Beginners-Guide/html/
- http://tldp.org/LDP/abs/html/
- http://wiki.bash-hackers.org/doku.php

NOKIA