

Operacijski sistemi

vaje 12

regularni izrazi

Metaznaki za tvorbo regularnih izrazov:

\ . ^ \$ | ? * + - : , () [] { }

znak

iskanje znaka/znakov (če ni metaznak)

^

začetek vrstice

\$

konec vrstice

\<beseda\>

niz beseda kot cela beseda (ne kot podniz)

\b

rob besede

\B

znotraj besede (ne rob besede)

regularni izrazi

Nabori znakov:

.

... en poljuben znak

[znaki]

... en znak iz nabora znakov (lahko uporabljamo okrajšave za zapis intervalov, npr. [a - z], [1 - 5])

[^znaki]

... en znak iz komplementa nabora znakov

preizkusimo

```
1. man grep | grep -C 100 "."
2. man grep | grep -C 100 "\."
3. man grep | grep -C 100 "\.$"
4. man grep | grep -C 100 "a."
5. man grep | grep -C 100 ".a."
6. man grep | grep -C 100 "\Bce"
7. man grep | grep -C 100 "ce\b"
8. man grep | grep -C 100 "the"
9. man grep | grep -C 100 "\<the\>"
10. man grep | grep -C 100 "[yYnN]"
11. man grep | grep -C 100 "\b[yYnN]"
12. man grep | grep -C 100 "[a-z]"
13. man grep | grep -C 100 "[^a-z]"
14. man grep | grep -C 100 "[a-zA-Z0-9]"
15. man grep | grep -C 100 "[-+0-9]"
16. man grep | grep -C 100 "[^aeiou]"
```

več načinov zapisa

	POSIX	vim, Perl	ASCII	opis
1.	<code>[:alnum:]</code>		<code>[A-Za-z0-9]</code>	alfanumerični znaki
2.		<code>\w</code>	<code>[</code>	alfanumerični znaki in <code>_</code>
3.		<code>\W</code>	<code>[^A-Za-z0-9_]</code>	vsi znaki, razen alfanumeričnih in <code>_</code>
4.	<code>[:alpha:]</code>	<code>\a</code>	<code>[</code>	črke
5.	<code>[:blank:]</code>		<code>[</code>	presledek in tabulator
6.	<code>[:cntrl:]</code>		<code>[\x00-\x1F\x7F]</code>	kontrolni znaki
7.	<code>[:digit:]</code>	<code>\d</code>	<code>[</code>	števke
8.	<code>^[[:digit:]]</code>	<code>\D</code>	<code>[</code>	vse, razen števk
9.	<code>[:graph:]</code>		<code>[\x21-\x7E]</code>	vidni znaki
10.	<code>[</code>	<code>\l</code>	<code>[a-z]</code>	male črke
11.	<code>[:print:]</code>	<code>\p</code>	<code>[\x20-\x7E]</code>	vidni znaki in presledek
12.	<code>[:punct:]</code>		<code>[\]\[!\"#\$%&'()*+,-./:;<=>?@^_`{ }~ -]</code>	ločila
13.	<code>[:space:]</code>	<code>\s</code>	<code>[\t\r\n\v\f]</code>	znaki za presledke različnih vrst
14.	<code>[</code>	<code>\S</code>	<code>[^ \t\r\n\v\f]</code>	vsi znaki, razen različnih presledkov
15.	<code>[:upper:]</code>	<code>\u</code>	<code>[</code>	velike črke
16.	<code>[:xdigit:]</code>	<code>\x</code>	<code>[</code>	šestnajstiške števke

Vstavimo manjkajoče zapise!

več načinov zapisa

	POSIX	vim, Perl	ASCII	opis
1.	<code>[:alnum:]</code>		<code>[A-Za-z0-9]</code>	alfanumerični znaki
2.		<code>\w</code>	<code>[A-Za-z0-9_]</code>	alfanumerični znaki in <code>_</code>
3.		<code>\W</code>	<code>[^A-Za-z0-9_]</code>	vsi znaki, razen alfanumeričnih in <code>_</code>
4.	<code>[:alpha:]</code>	<code>\a</code>	<code>[A-Za-z]</code>	črke
5.	<code>[:blank:]</code>		<code>[\t]</code>	presledek in tabulator
6.	<code>[:cntrl:]</code>		<code>[\x00-\x1F\x7F]</code>	kontrolni znaki
7.	<code>[:digit:]</code>	<code>\d</code>	<code>[0-9]</code>	števke
8.	<code>^[[:digit:]]</code>	<code>\D</code>	<code>[^0-9]</code>	vse, razen števk
9.	<code>[:graph:]</code>		<code>[\x21-\x7E]</code>	vidni znaki
10.	<code>[:lower:]</code>	<code>\l</code>	<code>[a-z]</code>	male črke
11.	<code>[:print:]</code>	<code>\p</code>	<code>[\x20-\x7E]</code>	vidni znaki in presledek
12.	<code>[:punct:]</code>		<code>[\]\[!"#\$%&'()*+,-./:;<=>?@\^_`{ }~ -]</code>	ločila
13.	<code>[:space:]</code>	<code>\s</code>	<code>[\t\r\n\v\f]</code>	znaki za presledke različnih vrst
14.	<code>^[[:space:]]</code>	<code>\S</code>	<code>[^ \t\r\n\v\f]</code>	vsi znaki, razen različnih presledkov
15.	<code>[:upper:]</code>	<code>\u</code>	<code>[A-Z]</code>	velike črke
16.	<code>[:xdigit:]</code>	<code>\x</code>	<code>[A-Fa-f0-9]</code>	šestnajstiške števke

ponavljanje

Regularni izraz	Pomen
$?$	nič ali ena ponovitev
$*$	nič ali več ponovitev
$+$	ena ali več ponovitev
$\{n\}$	natanko n ponovitev
$\{n, \}$	več ali enako n ponovitev
$\{n, m\}$	od n do m ponovitev

osnovni in razširjeni regularni izrazi

- osnovni regularni izrazi
 - stik ab , izbira $[ab]$
 - kvantifikacija $*$ $\{n\}$ $\{n, \}$ $\{n, m\}$
 - skupinjenje (ab)
- razširjeni regularni izrazi
 - stik ab , izbira $[ab]$ $a|b$
 - kvantifikacija $?$ $*$ $+$ $\{n\}$ $\{n, \}$ $\{n, m\}$
 - skupinjenje (ab) in povratni sklici $\backslash 1$

preizkusimo

1. man grep | grep -C 100 "[a-z][a-z]"
2. man grep | grep -E -C 100 "[a-z]{2}"
3. man grep | grep -E -C 100 "ti*"
4. man grep | grep -E -C 100 "(ti)*"
5. man grep | grep -E -C 100 "([tc])([ho])"
6. man grep | grep -E -C 100 "\b[A-Z][a-z]{2}"
7. man grep | grep -E -C 100 "\b[A-Z]?[a-z]{2}"
8. man grep | grep -E -C 100 "th.*s"
9. man grep | grep -E -C 100 "th[^s]*s"

naloge (1)

1. V pomoči ukaza `grep` poiščite vse nize, ki se začnejo s črko `A` ali `a`.
2. V datoteki `/etc/passwd` poiščite vse vrstice, kjer se pojavi beseda `admin` (ne podniz `admin`)
3. V datoteki `test.txt` poiščite vse vrstice, kjer so za besedilom na koncu vrstice presledki.

Kaj predstavljajo naslednji regularni izrazi (s kakšnimi nizi se ujema)?

4. `".{6}"`
5. `".{6,}"`
6. `"(19|20)[0-9]{2}"`
7. `"(+|-)?[0-9]+(\\.[0-9]+)?"`
8. `"(\\w+)\\1"`
9. `"\\<([^]+)[]+\\1"`
10. Kaj predstavlja naslednji regularni izraz (s kakšnimi nizi se ujema)? Izraz poenostavite tako, da ponavljanja zapišete kot ponavljanja.
`[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}`
11. Kaj predstavlja naslednji regularni izraz (s kakšnimi nizi se ujema)?
`^(0[1-9]|[12][0-9]|3[01])[-/.] (0[1-9]|1[012])[-/.] (19|20)[0-9][0-9]$`

Hierarhija Chomskega

jezik	gramatika (opis)	avtomat (realizacija)
Turingovi jeziki	gramatike brez omejitev (gramatika tipa 0)	Turingovi stroji
kontekstno odvisni jezik	kontekstno odvisne gramatike (gramatika tipa 1)	linearno omejeni avtomati
kontekstno neodvisni jezik	kontekstno neodvisne gramatike (gramatika tipa 2)	skladovni avtomat
regularni jezik	regularni izrazi (gramatika tipa 3)	končni avtomat

sed

- bere vrstico za vrstico
 - besedilo s standardnega vhoda
 - iz datoteke
- nad vsako vrstico izvede podane ukaze, npr.:
 - zamenjava: `'s/stari_niz/novi_niz/zastavice'`
 - brisanje: `'od_vrstice,do_vrstice d'`
 - brisanje: `'/vzorec_vrstice/ d'`
- izpisuje na standardni izhod
- uporabljamo lahko regularne izraze
 - pozna samo `?` in `*` (prepozna `+`, če poženemo z `-r`)
 - če se hočemo sklicevati na nek izraz, moramo tiste oklepaje označiti kot `\(` in `\)`

naloge (2)

Naslednje naloge rešite z uporabo ukaza `sed`.

1. Izpišite prvih 5 vrstic datoteke `/etc/passwd`.
2. V izpisu datoteke `/etc/passwd` zamenjajte pojavitve znaka `:` s tabulatorjem.
3. Z uporabo ukaza `sed` izpišite samo uporabniška imena iz datoteke `/etc/passwd`.
4. Z uporabo ukaza `sed` izpišite vse vrstice iz datoteke `/etc/passwd` razen vrstice uporabnika `student`.
5. Z uporabo ukaza `sed` v izpis datoteke `/etc/passwd` dodajte oklepaje okrog prve številke v vrstici/vseh številk.

naloge (2)

Kaj naredijo naslednji ukazi?

6. `sed '/^student:/ s/\home\student\home2\student/g' < /etc/passwd`
7. `sed '/^student:/ a\nejc:x:1003:1003:Nejc Gorenjc:/home/nejc:/bin/bash' < /etc/passwd`
8. `sudo cat /etc/shadow | sed '/^student:/ s/:\([^!]*\):/:\1:/' | tee izpis.txt`
9. `sed 's/\([^:]*\):\([^:]*\):\([^:]*\):\([^:]*\):\([^:]*\):/\5:\2:\3:\4:\1:/' < izpis.txt`
10. V datoteko `sed.cmd` napišite naslednje zaporedje ukazov `sed`:

```
10,16 s/u/a/g
10,16 s/o/u/g
10,16 s/i/o/g
10,16 s/e/i/g
10,16 s/a/e/g
16 i\PREDZADNJA VRSTICA
17,$ d
```

Kaj se zgodi z besedilom, nad katerim poženemo `sed` s podanim zaporedjem ukazov?