

VI editor - osnove korišćenja

pivo@bsd.org.yu

Sadržaj

1	Uvod	2
2	Pre nego što počnemo	2
3	Startovanje vi-a	3
4	Malo više o vi modovima (komandni mod i mod kucanja)	3
4.1	Mod kucanja (insert mod)	3
4.2	Komandni mod	4
4.2.1	Osnovna sintaksa komandnog moda	4
4.2.2	Komande kretanja	4
4.2.3	Komande kretanja po ekranu-skrolovanje	4
4.2.4	Razna taktička pomeranja :-)	4
4.2.5	Komande kretanje po tekućoj liniji a i šire :-)	4
4.2.6	Komande brisanja	5
4.2.7	Komande za zamenu teksta (replacing text)	5
4.2.8	Umetanje iz bafera (yank i paste operacija)	5
4.2.9	Komande pretraživanje teksta ili karaktera	5
4.2.10	Manipulacija karakterima i linijsko formatiranje	6
5	Razne korisne komande	6
6	Par korisnih saveta	6
7	Zaključak	6

1 Uvod

Vi je jedan od najmoćnijih programa za obradu (editovanje) fajla. Nastao 1976 godine na Berkliju (University of California at Berkeley), delo je William (Billy) Joy-a. Korisnici UNIX-a i programeri ga najčešće koriste. Ima tekstualni režim rada (nije grafički). Predstavlja drugo lice Ex-a (linijskog editora), radi se o istom programu ali sa drugačijim 'izgledom' (interfejsom). Mnogi početnici izbegavaju *vi* editor zato što se razlikuje od onih sa kojima se obično sreću. Ovo mini uputstvo je za one ljude koji su čuli za *vi* editor i hteli bi da savladaju neke osnove korišćenja. Pre nego što počnemo, mala napomena. Kada naletite na znak recimo `^ r` to znači da je pritisnuta tipka CTRL+r.

2 Pre nego što počnemo

Pošto *vi* koristi pun prikaz radnog ekrana, prvo treba da proverimo jel nam dobro podešena \$TERM promenljiva okruženja. Nema šta da brinete, najčešće je sve OK. Ako znate da koristite cons2512 emulaciju (ja koristim tu emulaciju na FreeBSD-u) terminala onda bi trebalo da na ukucano `echo $TERM` izbacite sledeće:

```
% echo $TERM
cons2512
```

Ako vam ne izbacite, a znate koju emulaciju koristite, onda ručno postavite vrednost za \$TERM. Naravno ovo podešavanje zavisi od ljuske (shell) koju koristite. Da bi ste videli koju ljusku koristite uradite ovo (kod mene je zsh):

```
% echo $SHELL
/usr/local/bin/zsh
```

Podešavanje emulacije terminala u zsh, ksh, bash je:

```
% TERM=cons2512
% export TERM
```

Za csh, tcsh je:

```
% setenv TERM=cons2512
```

Onda resetujemo terminal sa:

```
% tset
```

ili

```
% reset
```

NAPOMENA: Ako terminal koristimo u grafičkom okruženju (u X-u) onda odgovarajuća \$TERM varijabla je xterm-color (Na FreeBSD-u), znači:

```
% TERM=xterm-color
% export TERM
```

ili

```
% setenv TERM=xterm-color
```

Ako vam nijedna emulacija ne radi onda probajte sa sledećim vrednostima vt220 ili vt110.

3 Startovanje vi-a

VI editor se koristi za pravljenje ili obrađivanje tekst fajla. Komanda za startovanje vi editora je `vi`. Na primer želimo da obradimo fajl `obradi-me.txt` kucamo sledeće: `vi obradi-me.txt`. Kada se startuje vi sigurno će te primetiti mnoštvo tildi (ovako izgleda `~`) sa leve strane radnog ekrana. A na dnu ekrana primetiće te da stoji naziv fajla kao i još neke karakteristike pomenutog fajla (broj linija, da li je menjan i sl). Ako fajl ne postoji onda će na dnu ekrana pisati ovako nešto "obradi-me.txt" [New file]

Ako kojim slučajem ne dobijate ove kontrolne poruke pri startovanju moguće je da niste lepo podesili \$TERM promenljivu. Ako je tako, onda izađite iz vi editora, tako što će te ukucati `:q` (dve tačke-q-ENTER), i ponovo pročitajte drugo poglavlje.

Pošto znate da uđete u vi, najbolje je odmah da naučimo i da izađemo. Naime vi ima 2 režima (moda) rada. I to komandni (command mod) mod i mod kucanja (insert mod). Da bi smo došli u komandni mod. Dovoljno je pritisnuti tipku ESC. Ako se čuje pištanje, to znači da smo već u komadnom modu i ne treba više pritiskati ESC. Ako taj taster (ESC) nemamo na tastaturi, možemo preći u komandni mod sa kombinacijom dva tastera, i to sa `CTRL+I` (držimo istovremeno taster CTRL i I).

Komanda za izlazak, pod uslovom da ste u komandnom modu je `:q` (dve tačke-q-ENTER). Ako niste menjali tekući fajl, uspešno će te izaći. Međutim ako se u tom fajlu desila promena, nećete uspeti da izađete i moraće te da mu kažete: izađi na silu bez ikakvih promena! Ovako: `:q!` (dve tačke-q-znak uzvika-ENTER). Da ponovimo još jedanput. Ovim smo izašli iz vi-a a da nije zabeležena (snimljena) nikakva promena nad datim fajlom.

Normalno program za obradu fajlova ne bi bio to što jeste da ne može da snimi promenu obrađenog fajla. Snimanje promena u vi-u je (takođe moramo biti u komandnom modu) `:w` (dve tačke-w-ENTER). Ovim smo snimili promenu i dalje ostali u vi-u (nismo izašli iz programa). Ako želimo dati fajl da snimimo i pod drugim imenom onda `:w drugi-fajl.txt`. Ako želimo da snimimo promenu obrađenog fajla i izađemo iz editora (vi-a), kucamo sledeće: `:wq` (dve tačke-w-q i na kraju ENTER). Drugi način da snimimo promenu i izađemo iz editora je: kucamo ZZ (dvaput veliko Z, ali i za ovaj način moramo biti u komandnom modu). Ovaj način je jako zgodan za brzo snimanje i brz izlazak iz editora.

4 Malo više o vi modovima (komandni mod i mod kucanja)

4.1 Mod kucanja (insert mod)

Da bi smo mogli bilo šta da kucamo moramo ući u mod kucanja. Postoje sledeći načini ulaznja u taj režim rada:

- i uđi u mod kucanja sa pozicije kursora (kucanja)
- I uđi u mod kucanja na početku tekuće linije (kucanja)
- A uđi u mod kucanja na kraju tekuće linije
- a uđi u mod kucanja od mestu za jedno polje udesno od pozicije kursora
- O uđi u mod kucanja tako da tekuću liniju spustiš na dole za jedan red
- 0 uđi u mod kucanja tako da poziciju kursora spustiš na dole za jedan red

4.2 Komandni mod

4.2.1 Osnovna sintaksa komandnog moda

Komandni mod ima sledeću sintaksu (pravilo)

[koliko puta] [komanda] [gde]

[komanda] je obično jedan karakter (računajući i u kombinaciji sa CTRL). A [koliko] je broj (bilo koji broj). Na primer ako kucamo u komandnom modu 21x obrisćete 21 karakter nadesno od pozicije kursora. [gde] se odnosi na koliko linija,paragrafa i sl se odnosi data komanda.

4.2.2 Komande kretanja

h pomeranje za jedno u levo
j pomeranje za jedno na gore
k pomeranje za jedno na dole
l pomeranje za jedno u desno

Možete se i poslužiti i kursor tasterima (U novim verzijam vi ili vim-a. Od koje verzije je tako, ne znam).

4.2.3 Komande kretanja po ekranu-skrolovanje

CTRL+f skrolovanje celog ekrana napred
CTRL+b skrolovanje celog ekrana unazad
CTRL+u skrolovanje pola ekrana napred
CTRL+b skrolovanje pola ekrana unazad
CTRL+e skrolovanje jedne linije napred
CTRL+y skrolovanje jedne linije unazad

4.2.4 Razna taktička pomeranja :-)

M pomeri kursor na prvi neprazni karakter na sredini ekrana
H pomeri kursor na prvi neprazni karakter na vrh ekrana
L pomeri kursor na prvi neprazni karakter na dnu ekrana
+ pomeri kursor na prvi neprazni karakter u sledećoj (narednoj) liniji
- pomeri kursor na prvi neprazni karakter u prethodnoj (prošloj) liniji
{ pomeri kursor na početak paragrafa u odnosu na kursor
} pomeri kursor na kraj paragrafa u odnosu na kursor
nG idi na n-tu liniju (npr 23G ide na 23 liniju fajla)
G idi na zadnju liniju fajla
1G idi na prvu liniju fajla

4.2.5 Komande kretanje po tekućoj liniji a i šire :-)

\$ idi na kraj linije
0 idi na početak linije
w idi na početak sledeće reči
e idi na kraj reči u odnosu na trenutnu poziciju
E idi na kraj tekuće ili na kraj sledeće reči ali preskoči znake interpunkcije

b idi na početak tekuće ili početak prethodne reči
B idi na početak tekuće ili prethodnu reč ali preskoči znake interpunkcije
^ idi na prvu reč u tekućoj liniji
_ idi na prvi neprazni karakter prve reči u tekućoj liniji
| idi na zadatu [koliko] kolonu u tekućoj liniji (npr: 13|, ide na 13 kolonu)

4.2.6 Komande brisanja

dd briše celu tekuću liniju
d^ briše od kursora ne levo do početka linije
d0 radi isto što i d^
d\$ briše od tekuće pozicije kursora pa sve do kraja linije
D radi isto što i d\$
d} briše od tekuće pozicije pa sve do kraja paragrafa
x briše tekući karakter sa leve strane
X briše tekući karakter sa desne strane
dw briše celu reč od pozicije kursora na desno

Napomena: Kada se reč, pasus, slovo ili šta već obriše, to ostaje u baferu i može se iskoristiti za umetanje (copy/paste operacija).

4.2.7 Komande za zamenu teksta (replacing text)

C promeni sve do kraja linije od tekuće pozicije kursora
R zameni sve od pozicije kursoram, zamena se prekida sa tasterom ESC
S promeni celu liniju
cw promeni sve od pozicije kursora pa do kraja reči.
2cw promeni sve od pozicije kursora pa do kraja druge po redu reči
r zameni tekući karakter i vrati se u komandni mod
s zameni tekući karakter i ostani u mod kucanja (insert mod)

4.2.8 Umetanje iz bafera (yank i paste operacija)

yy na liniji gde je kursoro smesti celu liniju u bafer
Y isto radi što i yy
5yy od linije gde je kursor smesti 5 linija na dole u bafer
y3w smesta u bafer polja od pozicije kursora pa do treće reči
y} smesta u bafer ceo paragraf nagore od pozivije kursora.
P lepi sadržaj bafera na mestu kursora
p lepi zadržaj bafera na mesto od kursora na desno

4.2.9 Komande pretraživanje teksta ili karaktera

/ potraži znak ili reč koji je pored kose crte (sleša) na dole u fajlu
? potraži znak ili reč koji je pored kose crte (sleša) na gore u fajlu
F potraži karakter unazad u liniji koji prethodi posle ukucane komadne F
f potraži karakter napred u liniji koji sledi posle ukucane komande f
n potraži ponovo istu reč koja je zadata da se traži sa / ili ?
N radi isto što i ?
T nađi znak unazad u liniji i stani za jedno mesto udesno od nađenog
t nađi znak napred u liniji i stani za jedno mesto levo od nađenog

; ponovi zadnje komande zadate sa f, F, t ili T
, ponovi zadnje komande zadate sa f, F, t ili T u obrnutom redosledu
a[nb] potraži sve što počinja na slovo 'a' a posle njega sledi n ili b

4.2.10 Manipulacija karakterima i linijsko formatiranje

~ ako su mala slova promeni u velika na poziciji kursora i obrnuto
J spoji dve linije, tekuću i narednu u jednu liniju (jako česta opcija)
2>> dve linije (tekuću i narednu liniju) otpočeka pomera u desno [TAB]
3<< tri linije (tekuću i naredne dve linije) otpočeka pomera u levo [TAB]

5 Razne korisne komande

CTRL+g prikaži status info o fajlu (broj linija, ime fajla i sl)
CTRL+l osveži radni ekran
!ls pokreni ls komandu iz vi editora i prikaži ih
:r novi.txt uvuci u tekući fajl novi.txt koji se nalazi u tekućem dir-u
:r !ls umetni prikaz komande ls u fajlu
. ponovi zadnju komandu, kojom smo izvršili neko editovanje
u vrati zadnju promenu na staru (operacija UNDO)
U vrati zadnju linije u prethodno stanje pre zadnjeg obrade
:1,\$s/sr/SR/g gde se pojavljuje reč sr zameni sa SR (odnosi se na ceo fajl)
:set num postavi numerisanje linije u fajlu
:set nonum ukidanje numerisanje linija u fajlu
:ab ff/FreeBSD kada kucamo ff(mod kucanja) da nam ubaci FreeBSD umesto ff.

6 Par korisnih saveta

Ako želite da iz komandne linije ljuške (shell-a) odmah odete na 23 liniju fajla neki-kod.c. Možete uraditi ovako:

```
% vi +23 neki-kod.c
```

Ako želite da odmah u datom fajlu potražite reč while kucajte ovako (isto iz komandne linije (shell-a))

```
% vi +/while neki-kod.c
```

Ako vam je recimo nestala struja i želite da povratite ono što ste zadnje kucali a niste stigli da snimate promene, uradite sledeće:

```
% vi -r neki-kod.c
```

7 Zaključak

Priča o VI programu za obradu fajlova se nije završila na ovome, naprotiv tek smo započeli da otkrivamo mogućnosti ovog editora. Ovo može biti dobar početak da za dalje napredovanje u korišćenju VI editora.

Važno je shvatiti moć i fleksibilnost ovog editora, tj da se do raznih novih komandi dođe logičkim putem, a to je jedino moguće praksom i vežbom (nekima je dovoljno

samo jedan dan). Naravno tu je uvek uvek mučna strana manuala **man vi**, Od dokumentacije sa interneta vam preporučujem uputstvo koje je napisao tvorca vi-a Billy Joy.

<http://docs.FreeBSD.org/44doc/usd/12.vi/paper.html>

Iskreno se nadam se da će vam ovo uputstvo biti nekom od pomoći. Kao i to da će te sve uočene nepravilnosti bilo kakve vrste javiti autoru na mail pivo@bsd.org.yu radi ispravljanja istog.

pećko pivo