

# Otvoreni operativni sistemi

Marko Dimitrijević

# Tekstualni fajlovi

- Tekstualni fajlovi su obični fajlovi čiji je sadržaj kodiran ASCII, UNICODE ili nekim drugim kodom za kodiranje teksta.
- Tekstualni fajl se sastoji od **karaktera**. Karakter je vidljiv znak – slovo, cifra, interpunkcija, specijalni znak, ... Postoje kontrolni karakteri, koje nije moguće prikazati niti odštampati (CR, LF, BELL,...)
- Sadržaj tekstualnog fajla se može prikazati, odštampati, editovati, pretraživati, itd., u zavisnosti od prava pristupa korisnika.

# Tekstualni fajlovi

- Operacije sa tekstualnim fajlovima:
  - cat
  - more, less
  - head, tail
  - grep, egrep
  - cmp, diff
  - sort, uniq
  - wc
- Redirekcija
  - stdin
  - stdout
  - stderr

# Prikazivanje sadržaja tekstualnog fajla – cat

- Sadržaj tekstualnog fajla se može prikazati komandom **cat**:

**cat ime\_fajla**

- Opcije:

**cat -n ime\_fajla**

- ispisuje brojeve linija

**cat -E ime\_fajla**

- ispisuje \$ na kraju svake linije

**cat -T ime\_fajla**

- ispisuje ^I na mestu svakog tab karaktera

# Prikazivanje sadržaja tekstualnog fajla – more

- Sadržaj tekstualnog fajla se može prikazati komandom **more**:

**more ime\_fajla**

- Interaktivni mod:

**SPACE**

- pomera polovinu ekrana

**ENTER**

- pomera jednu liniju

**q**

- izlaz

**/**

- pretraga teksta

# Prikazivanje sadržaja tekstualnog fajla – less

- Sadržaj tekstualnog fajla se može prikazati komandom **less**:

**less ime\_fajla**

- Interaktivni mod:

**SPACE**

- pomera polovinu ekrana

**ENTER**

- pomera jednu liniju

**q**

- izlaz

**/**

- pretraga teksta

# Prikazivanje sadržaja tekstualnog fajla – head

- Početak (head) tekstualnog fajla se može prikazati komandom **head**:

## **head ime\_fajla**

- Opcije:

**head -n ime\_fajla** - ispisuje prvih n linija fajla

**head -c K ime\_fajla** - ispisuje prvih K bajtova binarnog fajla

# Prikazivanje sadržaja tekstualnog fajla – tail

- Kraj (tail) tekstualnog fajla se može prikazati komandom **tail**:

**tail ime\_fajla**

- Opcije:

**tail -n ime\_fajla**

- ispisuje poslednjih n linija fajla

**tail -c K ime\_fajla**

- ispisuje poslednjih K bajtova binarnog fajla

**tail -f ime\_fajla**

- ispisuje kraj fajla, i njegovu promenu



# Pronalaženje pattern-a u tekstualnom fajlu – grep

- Tekstualni šablon (pattern) u fajlu se može naći komandom **grep**:

**grep pattern ime\_fajla**

- Opcije:

**grep -i pattern ime\_fajla**

- ignoriše mala/velika slova

**grep -v pattern ime\_fajla**

- invertuje ispis

**grep -n pattern ime\_fajla**

- ispisuje broj linije fajla u kojoj se pattern nalazi

**grep -c pattern ime\_fajla**

- ispisuje samo broj linije fajla u kojoj se pattern nalazi

**grep -l pattern fajlovi**

- ispisuje samo imena fajlova u kojima se pattern nalazi

# Pronalaženje pattern-a u tekstualnom fajlu – egrep

- Tekstualni šablon (pattern) u fajlu se može naći komandom **egrep**. Pattern može da sadrži **regularne izraze**.

## **egrep pattern ime\_fajla**

- Opcije:
  - egrep -i pattern ime\_fajla** - ignoriše mala/velika slova
  - egrep -v pattern ime\_fajla** - invertuje ispis
  - egrep -n pattern ime\_fajla** - ispisuje broj linije fajla u kojoj se pattern nalazi
  - egrep -c pattern ime\_fajla** - ispisuje samo broj linije fajla u kojoj se pattern nalazi
  - egrep -l pattern fajlovi** - ispisuje samo imena fajlova u kojima se pattern nalazi

# Upoređivanje sadržaja tekstualnih fajlova - cmp

- Sadržaj tekstualnih fajlova se može uporediti komandom **cmp**:

```
cmp file_1 file_2
```

# Upoređivanje sadržaja tekstualnih fajlova - diff

- Sadržaj tekstualnih fajlova se može uporediti komandom **diff**:

```
diff file_1 file_2
```

- Opcije:

<b>diff -i file_1 file_2</b>	- ignoriše mala/velika slova
<b>diff -b file_1 file_2</b>	- ignoriše razlike u SPACE karakterima
<b>diff -E file_1 file_2</b>	- ignoriše razlike u TAB karakterima
<b>diff -B file_1 file_2</b>	- ignoriše razlike u praznim linijama
<b>diff -w file_1 file_2</b>	- ignoriše sve white space karaktere

# Sortiranje fajlova - sort

- Sadržaj tekstualnog fajla se može sortirati komandom **sort**:

**sort ime\_fajla**

- Opcije:

**sort -f ime\_fajla**

- ignoriše mala/velika slova

**sort -b ime\_fajla**

- ignoriše sve white space na početku

**sort -d ime\_fajla**

- sortira samo prema alfanumeričkom redosledu

**sort -M ime\_fajla**

- sortira po mesecima

**sort -h ime\_fajla**

- sortira human readable (1k, 2M, 3G,...)

# Pronalaženje i/ili izdvajanje linija koje se ponavljaju - uniq

- Linije koje se ponavljaju u fajlu mogu se izbeći komandom **uniq**:

**uniq -u ime\_fajla**

- Opcije:

**uniq -i ime\_fajla**

- ignoriše mala/velika slova

**uniq -c ime\_fajla**

- speciicira broj ponavljanja linije

**uniq -D ime\_fajla**

- ispisuje grupisane sve linije koje se ponavljaju

## Broj linija, reči, karaktera – wc

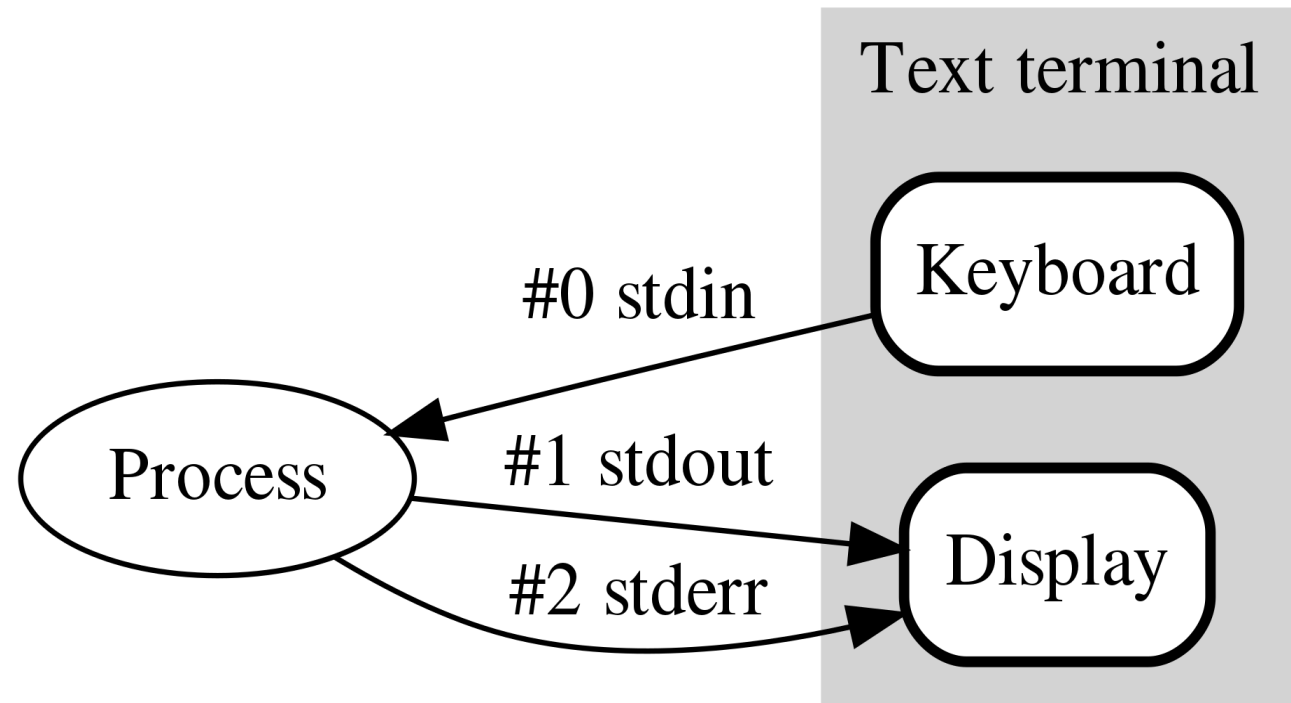
- Broj linija, reči ili karaktera tekstualnog fajla se mogu odrediti komandom **wc**:

<b>wc -l ime_fajla</b>	- broj linija u fajlu
<b>wc -w ime_fajla</b>	- broj reči u fajlu
<b>wc -c ime_fajla</b>	- broj karaktera u fajlu

# Redirekcija ulaza i izlaza - stdin, stdout i stderr

- Svi procesi koji se izvršavaju na UNIX/Linux sistemu imaju tri komunikaciona kanala:

1. **Standardni ulaz – stdin**
2. **Standardni izlaz – stdout**
3. **Dijagnostički izlaz – stderr**





## Standardni ulaz - stdin

- Standardni ulaz je mehanizam kojim se programu prosleđuju podaci – najčešće tekst.
- Proces prihvata podatke iz tekstualnog fajla, prilikom pozivanja komande (cat, grep, more, less, head, tail, wc, sort, uniq,...), preko tastature ili redirekcijom od drugog procesa.
- Nemaju sve komande standardni ulaz (cd, pwd, ls,...).

## Standardni izlaz – stdout

- Standardni izlaz je komunikacioni kanal preko koga proces šalje podatke.
- Neki procesi su nemi, ukoliko je izvršavanje uspešno, ne daju nikakav podatak.
- Standardni izlaz procesa se nasleđuje od parent procesa, tj. najčešće od shell-a (bash), i ispisuju podatke na terminalu (monitoru).

# Dijagnostički izlaz

- Dijagnostički izlaz je drugi komunikacioni kanal preko koga proces šalje podatke.
- Preko dijagnostičkog izlaza se šalju poruke o greškama ili druge dijagnostičke poruke. Na taj način je moguće razdvojiti standardni izlaz od poruka o greškama.
- Ukoliko nije drugačije specificirano, dijagnostički izlaz se prikazuje kao i standardni izlaz na terminalu.

# Redirekcija standardnog i dijagnostičkog izlaza

Komanda	Objašnjenje
<code>komanda &gt; file_name</code>	Preusmerava standardni izlaz u <b>file_name</b> . Ako <b>file_name</b> postoji gubi se prethodni sadržaj. Ovo se može sprečiti ako se setuje shell promenljiva <b>noclobber</b> ( <b>set noclobber</b> ). Primer:  <code>% cat file1 file2 &gt; file3</code>
<code>komanda &gt;! file_name</code>	Forsirano preusmeravanje. Iako <b>file_name</b> postoji i setovana je promenljiva <b>noclobber</b> , vrsi se prepisivanje starog sadržaja.
<code>komanda &gt;&amp; file_name</code>	Preusmeravanje standardnog izlaza i dijagnostičkog izlaza. Primer:  <code>% rm -r nonexisting_dir &gt;&amp; /dev/null</code>
<code>komanda &gt;&amp;! file_name</code>	Forsirano preusmeravanje standardnog i dijagnostičkog izlaza.

# Redirekcija standardnog i dijagnostičkog izlaza

Komanda	Objašnjenje
<code>komanda &gt;&gt; file_name</code>	Preusmeravanje standardnog izlaza u <b>file_name</b> . Zapis se dodaje već postojećem sadržaju datoteke <b>file_name</b> .
<code>komanda &gt;&gt;&amp; file_name</code>	Preusmeravanje standardnog i dijagnostičkog izlaza u <b>file_name</b> . Stari sadržaj se ne briše već mu se dodaje novi.
<code>komanda &gt;&gt;&amp;! file_name</code>	Forsirano preusmeravanje standardnog i dijagnostičkog izlaza u <b>file_name</b> . Stari sadržaj se ne briše.
<code>komanda 1&gt; file_1 2&gt; file_2</code>	Preusmeravanje standardnog izlaza u <b>file_1</b> i dijagnostičkog izlaza u <b>file_2</b> .
<code>komanda 1&gt;&gt; file_1 2&gt;&gt; file_2</code>	Preusmeravanje standardnog izlaza u <b>file_1</b> i dijagnostičkog izlaza u <b>file_2</b> . Zapis se dodaje već postojećim sadržajima fajlova <b>file_1</b> i <b>file_2</b> .

# Redirekcija standardnog izlaza na standardni ulaz

Komanda	Objašnjenje
<code>cmd1   cmd2   cmd3</code>	<p><b>Pipelining.</b> Standardni izlaz prethodne komande se usmerava na standardni ulaz naredne. Primer:</p> <pre>% ps -ef   grep getty   more</pre>
<code>(cmd1; cmd2;... ) &gt;&amp; file_name</code>	<p>Usmeravanje standardnog i dijagnostičkog izlaza grupe komandi u datoteku <b>file_name</b>. Primer:</p> <pre>% (ls -l; ls -a; ls -s) &gt;&amp; listings</pre>
<code>komanda &lt; file_name</code>	<p>Preusmeravanje standardnog ulaza iz fajla <b>file_name</b>. Primer:</p> <pre>% mail &lt; pismo</pre>
<code>komanda &lt;&lt; EOF</code>	<p>Saopstava se komandi <b>komanda</b> (obično je interaktivna) da standardni ulaz čita iz istog fajla. Primenjuje se u skriptovima. Reč <b>EOF</b> se tretira kao end-of-input.</p>