

Bash II

Eliška Jégrová

13. 10. 2025

Obsah

Roury a filtry

Echo

Hledání souborů a textu

Roury a filtry

- ▶ Unixový shell umí spojovat příkazy dohromady.
- ▶ Každý příkaz má standardní vstup a výstup.
- ▶ Např. jak bychom zjistili pomocí počítače, který soubor má nejvíce řádků a který nejméně?

```
$ cd ~/Dokumenty/data-shell/molecules
```

```
$ wc -l *.pdb
```

```
20 cubane.pdb
```

```
12 ethane.pdb
```

```
9 methane.pdb
```

```
30 octane.pdb
```

```
21 pentane.pdb
```

```
15 propane.pdb
```

```
107 celkem
```

* - odpovídá libovolnému počtu znaků

Přesměrování výstupu

- ▶ Výstup se standardně vypisuje do terminálu.



- ▶ Pomocí > lze přesměrovat výstup příkazu do souboru.



```
$ wc -l *.pdb > delky.txt  
$ cat delky.txt
```

Pozor: pokud soubor existuje, jeho obsah se přepíše.

Hledání nejmenší molekuly

- ▶ Řazení: `sort`
 - ▶ Přepínač `-n` zapíná číselné řazení.

```
$ sort delky.txt  
$ sort -n delky.txt
```

- ▶ Jak najít nejmenší molekulu?

```
$ wc -l *.pdb > delky.txt  
$ sort -n delky.txt > delky-serazene.txt  
$ head -n1 delky-serazene.txt  
9 methane.pdb
```

Přesměrování vstupu

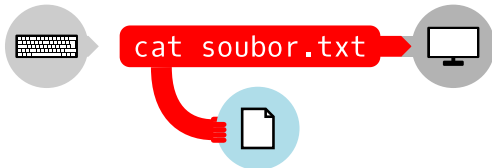
\$ cat



\$ cat < delky.txt



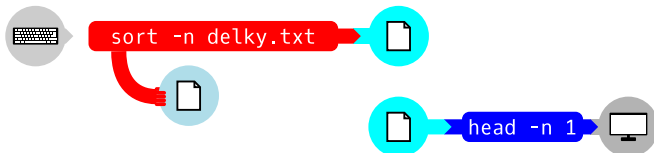
\$ cat delky.txt



Roury (pipes)

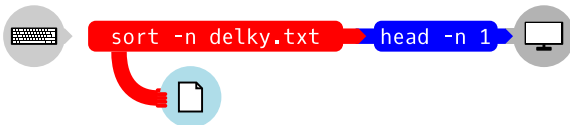
```
$ sort -n delky.txt > delky-serazene.txt
```

```
$ head -n1 < delky-serazene.txt
```

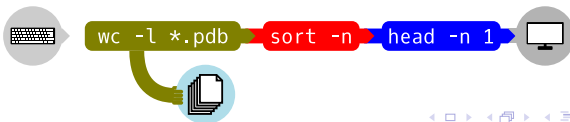


- Pomocí | přesměruj výstup jednoho příkazu na vstup druhého.

```
$ sort -n delky.txt | head -n1
```



```
$ wc -l *.pdb | sort -n | head -n 1
```



Samostatná práce – Analýza molekul

- ▶ Najdi nejdelší molekulu ve složce
~/Dokumenty/data-shell/molecules

1. Pomocí přesměrování do souboru

- ▶ Spočítej počet řádků všech .pdb souborů a ulož do delky.txt.
- ▶ Seřaď soubor delky.txt číselně a ulož do delky-serazene.txt.
- ▶ Vypiš poslední řádek souboru – to je nejdelší molekula.

2. Pomocí roury (pipe)

- ▶ Spočítej počet řádků a hned seřaď číselně a vypiš poslední řádek (jednořádkový příkaz)
- ▶ Najdi tři nejkratší molekuly (bonus)
– vyber si způsob jak úkol udělat

Příkaz echo

- ▶ Slouží k výpisu textu do terminálu.

```
$ echo Ahoj světe!  
Ahoj světe!
```

```
$ echo Ahoj      světe!  
Ahoj světe!
```

```
$ echo 'Ahoj      světe!'  
Ahoj      světe!
```

- ▶ Jednoduché vs Dvojité uvozovky

```
$ echo $0  
bash
```

```
$ echo "$0"  
bash
```

```
$ echo '$0'  
$0
```

Příkaz echo

- ▶ Echo do souboru - přepíše se soubor

```
$ echo haló haló > basnicka.txt
```

```
$ cat basnicka.txt
```

```
haló haló
```

- ▶ Echo do souboru - přidat na konec souboru

```
$ echo haló haló >> basnicka.txt
```

```
$ cat basnicka.txt
```

```
haló haló
```

```
co se stalo?
```

Samostatná práce – Echo a soubory

1. Vytvoř soubor `bajka.txt` a napiš do něj první řádek textu „Moje první báseň“ pomocí `echo` a přesměrování `>`.
2. Přidej další dva řádky s textem „Druhý řádek“ a „Třetí řádek“ na konec souboru pomocí `>>`.
3. Zobraz celý obsah souboru a ověř, že obsahuje tři řádky.
4. Bonus: Přidej jeden řádek, kde budeš mít víc mezer mezi slovy, a ověř, že se mezery zachovají při výpisu.

Příkaz grep

- ▶ Slouží k hledání řetězce v souborech.
- ▶ Základní syntaxe: `grep <výraz> <soubor>`
\$ `cd ~/Dokumenty/data-shell/writing`
\$ `grep se haiku.txt`
– vypíše všechny řádky obsahující výraz "se"
- ▶ Přepínače:
 - ▶ `-i` – ignoruje velikost písmen
 - ▶ `-n` – zobrazí čísla řádků
 - ▶ `-w` – hledá jen celá slova

Příkaz find

- ▶ Slouží k hledání souborů v adresářové struktuře.
- ▶ Základní syntaxe: `find <cesta> <kritéria>`
`$ find . -name '*.txt'`
– najde tyto soubory v aktuálním adresáři a podsložkách
- ▶ Stejný výsledek se dá získat pomocí `grep`
`$ ls -recursive | grep txt`

Samostatná práce – Hledání textu a souborů

- ▶ V adresáři ~/Dokumenty/data-shell/writing najděte všechny soubory .md.
 - ▶ Použijte příkaz `find`

Pro následující úkoly použijte soubor
data-shell/writing/data/LittleWomen.txt

- ▶ V souboru najděte všechny řádky obsahující slovo `Unlike`.
 - ▶ Použijte `grep`.
 - ▶ Vyzkoušejte přepínač `-i` pro ignorování velikosti písmen.
- ▶ Vypište čísla řádků, kde se vyskytuje slovo `reason`.
 - ▶ Použijte `grep -n`.
- ▶ Najděte všechny řádky obsahující celá slova `echo`.
 - ▶ Použijte přepínač `-w`.