

# Programování v unixovém shellu - úkoly

## Seminář 1

---

Spustěte aplikaci emulace terminálu (*Terminál*, *Konsole* apod.). Následující provádějte v této aplikaci (terminálu).

Napište do příkazové řádky terminálu jakýkoliv text, pohybujte po něm kurzorem (klávesy `←` a `→`, `C-←` a `C-→`, `Home`, `End`, `C-a`, `C-e`), editujte jej (klávesy `Del`, `C-d`, `Backspace`, `C-t`, `C-k`, `C-y`, označení textu myší a `S-C-c`, `S-C-v`) a potvrďte (klávesa `Enter`).

Prohlédněte si manuálovou stránku programu **date** (po zobrazení klávesy `↑` a `↓`, `PgUp`, `PgDn`, `h`, `q`).

Pomocí programu **date** (a jeho manuálové stránky) zobrazte (do terminálu) název dne v týdnu (pouze).

Pomocí programu **ssh** se připojte (přihlašte) pod svým uživatelským jménem na server `phoenix.inf.upol.cz` a odhlašte se z něj.

Listujte historií zadaných příkazů (klávesy `↑` a `↓`, `M-<`, `M->`), vyhledejte příkaz z historie (klávesa `C-r` a zadávání textu, `C-g`).

Zobrazte cestu k aktuálnímu adresáři.

Zobrazte seznam všech souborů a podadresářů, včetně tzv. skrytých, v aktuálním adresáři (= obsah adresáře) s podrobnějšími informacemi (jméno, velikost, datum, práva aj.) o nich.

Vytvořte adresář `tmp`, vejďte do něj a vyjďte z něj do nadřazeného adresáře.

Zobrazte nápovědu shellu k příkazu **cd** (program **help**).

Pomocí (grafického) programu **gedit** (*Textový editor*) upravte obsah souboru `welcome.html` v adresáři `~/public_html` (na serveru `phoenix.inf.upol.cz` je tento soubor interpretován jako webová stránka na adrese <http://phoenix.inf.upol.cz/~login>, kde `login` je uživatelské jméno).

Zobrazte obsah souboru `welcome.html` (programy **cat**, **less**, klávesy `↑` a `↓`, kolečko myši, `PgUp`, `PgDn`, `S-PgUp`, `S-PgDn`, `h`, `q`).

Zkopírujte soubor `welcome.html` do souboru `index.html`, ten přejmenujte na `main.html` a poté tento smažte (s volbami `-i` a `-f`). Zkopírujte adresář `public_html` na adresář `web` a poté tento smažte.

Vytvořte symbolický odkaz `index.html` na kopii souboru `welcome.html`, kopii smažte, opět vytvořte a pak smažte odkaz. Mezi kroky si zobrazte obsah adresáře s odkazem a soubory s podrobnějšími informacemi (o odkazu).

Zobrazte skupiny, do kterých („vás“) uživatel patří, a obsah adresáře s informacemi o právech souborů a podadresářů.

Odeberte souboru `welcome.html` právo zápisu a zkuste do něj zapsat (např. programem **gedit**), právo čtení a zobrazit jeho obsah, poté práva souboru vraťte.

Odeberte adresáři `public_html` právo vstupu (všem uživatelům) a zkuste zobrazit jeho obsah a vstoupit do něj, poté právo vraťte a odeberte právo čtení a opět zkuste zobrazit jeho obsah a vstoupit do něj, pak opět právo vraťte.

Nastavte všechna práva vlastníkově a pouze právo vstupu skupině a ostatním uživatelům adresáři `~/public_html`.

Zkuste v adresáři `/tmp` vytvořit soubor a smazat cizí soubor.

Připojte USB flash disk, zobrazte informaci o obsazeném a volném místě na něm (program **df**) a velikosti jednotlivých adresářů na něm (program **du**).

Zobrazte výpis všech procesů „vašeho“ a jiného uživatele, všechny procesy v systému, ve stromové struktuře, pouze s informací o PID a příkazu procesu.

Spust'te (grafický) program **gnome-calculator**, získejte PID jeho procesu a programem **kill** jej pozastavte (a zkuste jej ovládat), rozběhněte a ukončete. Spust'te jej poté znovu, z terminálu, a zavřete okno terminálu. A znovu, a ukončete shell v terminálu (jako odhlášení se např. v **ssh**).

Spust'te (grafické) programy **gnome-calculator** a **gnome-mines** (v terminálu), první tzv. na pozadí a druhý tzv. na popředí. Druhý přesuňte na pozadí a první na popředí.

Zobrazte seznam všech souborů a podadresářů v adresáři `/dev` se jménem začínajícím `ttycifra`, kde `cifra` je 0 až 9, a `ttynecifra`, kde `necifra` není 0 až 9.

Vytvořte soubory (pomocí programu **touch**) se jmény `***` a `"1 & 2"` (včetně obou `"`).

Při zadávání jmen programů a souborů nebo adresářů stiskněte (opakovaně) klávesu Tab!

Uložte výstup programu **date** do souboru a pak do něj přidejte výpis všech procesů v systému.

Zobrazte výpis všech procesů v systému v programu **less**.

## Seminář 2

---

Vytvořte v textovém editoru (**vi**, **vim**, **emacs**, **jed**, **nano** aj.) tabulkový soubor s několika řádky a sloupci (oddělenými např. tabulátorem). Vyzkoušejte si operace smazání (znaků a celého řádku), kopírování a vložení (znaků a celého řádku), undo, vyhledávání, náhrady řetězce za jiný.

Zobrazte očíslované řádky vstupu od 10. řádku do 20. řádku včetně v opačném pořadí řádků.

Setříd'te (tabulkový) výstup programu **df** podle čtvrtého sloupce (volné místo) číselně sestupně.

Prohod'te v tabulkovém vstupu (sloupce oddělené např. tabulátorem) první a druhý sloupec.

Zobrazte obsah adresáře s pouze informacemi o právech, velikosti a jménu.

Zobrazte počet skupin, do kterých („vás“) uživatel patří.

Zobrazte na jednom řádku seznam všech uživatelů, pod kterými běží v systému alespoň jeden proces.

Zobrazte pouze řádky souboru, které nejsou v jiném souboru, tj. „rozdíl“ souborů (jako rozdíl množin).

Zobrazte pouze řádky vstupu obsahující číslo zapsané v šestnáctkové soustavě začínající 0x (s malými i velkými písmeny, jako jedno slovo).

Zobrazte pouze řádky vstupu, které neobsahují dvě stejná čísla.

## Seminář 3

---

Vypište seznam všech souborů a podadresářů v adresáři zadaném jako argument (= obsah adresáře) s informacemi o typu (soubor, adresář, symbolický odkaz) a právech (čtení, zápis, spouštění) pro spouštějícího uživatele. Při žádném argumentu v aktuálním adresáři, při prvním argumentu -a (adresář by byl druhý) včetně tzv. skrytých souborů a podadresářů (jinak ne).

**[2 body]**

Naprogramujte „hádání“ (celého) čísla, které si uživatel myslí, z intervalu zadaného až dvěma argumenty, sérií dotazů „je menší/větší než X?“ algoritmem půlení intervalu. Při jednom argumentu je první číslo 0, při žádném navíc druhé 100.

**[2 body]**

Implementujte zjednodušenou verzi programu **seq**: výpis posloupnosti (celých) čísel oddělených mezerou od čísla zadaného jako první argument do čísla zadaného jako třetí argument, obojí včetně, s přírůstkem zadaným jako druhý argument – kladným, pokud je počáteční číslo menší nebo rovno než cílové, jinak záporným. Při dvou argumentech je chybějící přírůstek roven 1, při jednom je i chybějící počáteční číslo rovno 1.

**[4 body]**