

- MACSIMUS (též na VŠCHT serveru) nejlépe pracuje pod linuxem
- Všechny úlohy obsahují podrobný návod – stačí pouze základní znalosti příkazového řádku, viz [Linux pro začátečníky](#) dále
- Můžete pracovat na svém počítači. Výběr možností:
 - Na Windows 11 si [instalují WSLg \(doporučeno\)](#) nebo WSL (na Win 10) a [Xming](#)
 - Instalují si linux (např. nějakou verzi [Ubuntu](#)) místo svého OS, o který tak přijdu
 - Instalují si linux pomocí [dual-boot](#) zároveň s mým OS (vhodné pro geeky)
 - Instalují si linux ve [virtuálním počítači pod Windows](#) (lze i pro macOS, náročnější na počítač)
 - Pod Windows použijí port [MinGW](#) obdobné porty existují i pro macOS, jen s nimi nemám zkušenost
 - Pod Windows použijí port [CygWin](#)
- Můžete [pracovat na počítačích Ústavu fyzikální chemie](#). Možnosti:
 - Budu sedět ve škole
 - Budu uživatel [guest](#) (o heslo si řeknu)
 - Budu pracovat z domova
 - Mám vlastní účet na klastru
- Instalace a základní návod pro MACSIMUS

- **Operační systém** (OS) je systémový software, který ovládá hardware počítače a softwarové zdroje a poskytuje služby počítačovým programům.
- **Jádro** je centrální část operačního systému, která obecně řídí vše v systému. Je vždy (rezidentní) v paměti a stará se o interakci mezi hardwarem a softwarem.
- **Aplikace** je software provádějící specifické funkce pro koncového uživatele či pro další aplikaci.

Příklady aplikací:

Systémové grafické prostředí

Shell (příkazový řádek)

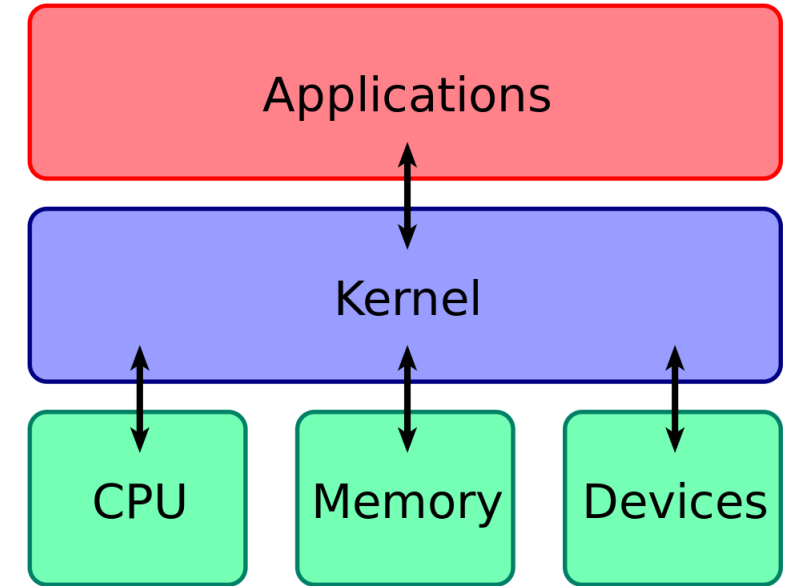
- Webový prohlížeč

- Webová aplikace (např. hra)

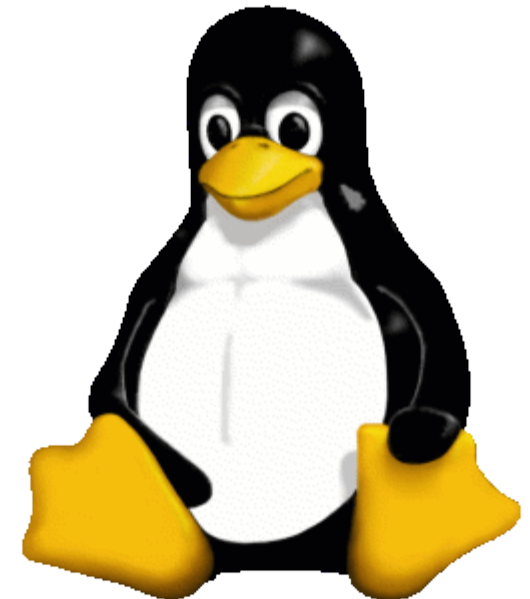
- Zoom / MS Teams

- Souborový manažer (průzkumník souborů)

- Zoom / MS Teams (samostatně)



credit: Wikipedia



Windows

- Proprietární, placený
- CP/M → DOS → Windows 3.1 → Windows 95 → Windows XP → Windows Vista → Windows 11
- Střední míra uživatelského přizpůsobení (nejmenší má iOS)
- Verze: „domácí“, „profesionální“, „server“
- Jeden systém distribuce softwaru
- Jedno grafické prostředí (window manager)
- Příkazový řádek: PowerShell, cmd; existují alternativy; registry
- Důraz na grafické prostředí

Linux

3/50
pchlm

- Open source, zadarmo vč. spousty softwaru + proprietární drivery a aplikace
- PDP-7 → Unix → BSD Unix → (MINIX) → Linux (Linus Torvalds)
- Vysoká míra uživatelského přizpůsobení
- Distro: stovky pro různá použití a úroveň uživatele (obvyklejší skupiny distribucí jsou: **Ubuntu**, Debian, Arch, Mint, OpenSUSE, Fedora, Red Hat, Mandriva, Slackware, Gentoo...; jádro: Android, ChromeOS)
- Systémy distribuce softwaru: **.deb**, **.rpm**, snap,...
- Grafická prostředí (window manager+): Gnome, **GNOME 2**, KDE, ...
- Příkazový řádek: **bash**, sh, csh, tcsh, ksh, zsh ...
- Lze provozovat bez grafického prostředí vč. startu grafických aplikací (srov. DOS/Win3.1)

● Doporučená distribuce: **Ubuntu Mate**

● Doporučený shell: **bash**

Spuštění z grafického prostředí:

- `Ctrl-Alt-T` příp. `Win-T`
- Pravé tlačítko myši na plochu → `Terminal`, `Open in Terminal` ...
- Najdeme na liště (panelu) 
- Najdeme v menu → `System Tools` → `MATE Terminal` (dle distribuce)
- `Ctrl-Alt-F1`, `Ctrl-Alt-F2`, ... = bez grafického prostředí, normálně se nedoporučuje;
`Ctrl-Alt-F7` = zpět grafické prostředí

Spuštění terminálu z Windows na vzdáleném počítači:

- **MobaXterm** (má integrovanou grafiku, ale občas nezvládne víc otevřených grafických oken)
- **PuTTY** (příkazový řádek) + **Xming** (grafika) – existuje i pro Mac

Příkazy v terminálu píšeme za “prompt”. Ten může vypadat různě.

```
jiri@403-a324-01:~ $
```

Zde použijeme `guest@403-a324-01:~/Videos $`, kde:

`guest` = uživatel, `403-a324-01` = jméno počítače, `~/Videos` = pracovní složka

Napíšeme a odešleme pomocí `Enter`.

`~ = /home/guest`

- Kdo jsem?

```
guest@403-a324-01:~ $ whoami
```

- Ve které složce jsem?

```
guest@403-a324-01:~ $ pwd
```

- Co je v této složce?

```
guest@403-a324-01:~ $ ls
```

```
guest@403-a324-01:~ $ ls -l
```

- Který soubor jsem naposledy změnil/a?

```
guest@403-a324-01:~ $ ls -rtl
```

- Jak se používá `ls`?

```
guest@403-a324-01:~ $ man ls
```

- Co je ve složce („adresáři“, *directory*) `sim`?

```
guest@403-a324-01:~ $ ls sim
```

nebo změním aktivní složku a pak se do ní podívám:

```
guest@403-a324-01:~ $ cd sim
```

```
guest@403-a324-01:~/sim $ ls
```

- Něco se stalo, na terminálu je zmatek: `Ctrl-C`

```
guest@403-a324-01:~ $ reset
```

- Jsem líný psát znova příkazy: kurzorové šipky

```
guest@403-a324-01:~ $ history
```

- tabulátor doplňuje dle kontextu, příklady:

```
guest@403-a324-01:~ $ histTab
```

```
guest@403-a324-01:~ $ ls DocTab
```

Ve výpisu `ls -l` znamená:

(např.: `drwxrwxr-x`)

d=directory (1. znak) a pak po

trojicích (user, group, all):

x: soubor lze spustit, složku otevřít

r: povoleno čtení

w: povolen zápis

- Pokud jste na sdíleném účtu **guest**: vytvoření nové složky, nastavení jako pracovní a kontrola:

za **JOSEFK** dosad'te vaše školní ID malými písmeny

```
guest@403-a324-01:~ $ cd # tím se vrátím „domů“ (za „# “ je komentář)
```

```
guest@403-a324-01:~ $ mkdir JOSEFK
```

```
guest@403-a324-01:~ $ cd JOSEFK
```

```
guest@403-a324-01:~/JOSEFK $ pwd  
/home/guest/JOSEFK
```

- Zvláštní názvy:

Domovská složka = **~**

Tato složka = **.**

Rodičovská složka = **..**

```
guest@403-a324-01:~/JOSEFK $ cd ..
```

```
guest@403-a324-01:~ $ pwd
```

cd ~ se dá zkrátit na **cd**

- Vše začíná v „kořenovém adresáři“ (*root*), absolutní cesta (*path*):

```
guest@403-a324-01:~ $ ls /
```

- Externí disky apod. jsou obvykle v **/media/USER/**

zkuste příkazy:

```
cd  
ls  
ls .  
cd ..  
ls  
pwd  
ls /home  
cd
```

Příkazy lze oddělovat středníkem:

```
ls;ls -l
```

Nejsou tu “drives” jako **A:** **B:** ve Windows

Ctrl-C	přerušení (lze programátorsky ovládat)
Ctrl-Shift-C	kopírovat do schránky (<i>clipboard</i>)
Ctrl-Shift-V	vložit (<i>paste</i>)
Ctrl-Shift-X	vyjmout (zkopírovat a smazat)
Ctrl-D	konec souboru (vkládaných dat)
Ctrl-Z	dej běžící aplikaci do pozadí (terminál se uvolní) zpět do popředí příkazem fg
Ctrl-S	zakaž výstup na terminál (může se stát omylem)
Ctrl-Q	povol výstup na terminál
Ctrl-R	interaktivní historie
Ctrl-I	= Tab , kontextové doplňování

- Ve většině “oknových” aplikací fungují **Ctrl-C**, **Ctrl-V**, **Ctrl-X** jako obvykle.
- Na některých (vzdálených) terminálech to může být jinak.
- Lze také jen myší označit a kopírovat prostředním tlačítkem (na vzdálených terminálech může být prostřední tlačítko).

Příklad:

```
guest@403-a324-01:~ $ cd JOSEFK
```

```
guest@403-a324-01:~/JOSEFK $ while true; do sleep 1; echo cheers; done
```

Zkuste **Ctrl-S**, **Ctrl-Q**, **Ctrl-C**

<code>exit</code>	konec/odhlášení (někdy též Ctrl-D)
<code>ls</code>	vypíše obsah složky
<code>ls -l *.pdf</code>	vypíše detailně seznam přítomných PDF souborů
<code>cd SLOŽKA</code>	změna pracovní složky (<i>change directory</i>)
<code>pwd</code>	vypíše absolutní cestu k pracovní složce (<i>print working directory</i>)
<code>mkdir SLOŽKA</code>	vytvoří prázdnou složku (adresář)
<code>rmdir SLOŽKA</code>	smaže prázdnou složku
<code>cp SOUBOR1 SOUBOR2</code>	kopíruje soubor; existuje-li SOUBOR2, je přepsán
<code>cp -i SOUBOR1 SOUBOR2</code>	kopíruje soubor; existuje-li SOUBOR2, zeptá se, zda má přepsat
<code>cp SOUBOR[y] KAM/</code>	kopíruje soubor(y) do složky; / na konci je dobrý zvyk
<code>cp -r COKOLIV KAM/</code>	kopíruje vše (u složek i obsah) do složky
<code>mv SOUBOR1 SOUBOR2</code>	přejmenuje soubor (nebo složku)
<code>mv COKOLIV KAM/</code>	přesune soubory/složky
<code>rm SOUBOR[y]</code>	smaže soubor(y)
<code>rm -i SOUBOR[y]</code>	zeptá se, má-li smazat soubory
<code>rm -r COKOLIV</code>	smaže vše (i složky)

<code>man PŘÍKAZ</code>	návod k příkazu PŘÍKAZ
<code>head SOUBOR</code>	vypíše začátek souboru/ů na obrazovku, <code>head -n3</code> jen 3 řádky
<code>tail SOUBOR</code>	vypíše konec souboru/ů na obrazovku, <code>tail -n3</code> jen 3 řádky
<code>ps</code>	vypíše běžící programy (procesy), všechny: <code>-e</code> , bez terminálu: <code>-x ...</code>
<code>top</code>	“task manager”: běžící procesy a alokace zdrojů, vypnout <code>q</code>
<code>kill PID</code>	pošle procesu signál <code>Ctrl-C</code> = „přeruš“ (lze ignorovat), PID viz <code>ps</code> / <code>top</code>
<code>kill -9 PID</code>	pošle procesu signál “vypni” (nelze ignorovat)
<code>fgrep SLOVO</code>	hledá SLOVO nebo jeho část (filtr)
<code>fgrep SLOVO SOUBOR</code>	hledá SLOVO v souboru/souborech
<code>grep SLOVO</code>	hledá tzv. regulární výraz (nedoporučeno pro začátečníky)
<code>ssh USER@COMP</code>	připojí vzdáleně uživatele USER na počítač (jméno / IP adresa) COMP
<code>ssh -X USER@COMP</code>	... včetně grafiky (X11)

Příklad: (řádky souboru `sim.def` jména obsahující nastavení teploty 'T=')

```
guest@403-a324-01:~ $ fgrep T= jmena.txt
```

nebo

```
guest@403-a324-01:~ $ fgrep T= < jmena.txt
```

Pozor, ve výchozím nastavení se `cp`, `mv`, `rm` neptají a soubor je **definitivně pryč!**

Náprava:

- Používat koš – lze jen, jste-li v grafickém prostředí:
 - v okénkovém prostředí (“Move to Trash”)
 - příkaz `trash-put` z balíku `trash-cli`
- Vždy používat `-i` (= interaktivně = slušně se zeptá)
- Aliasovat `cp` na `cp -i` (resp. `rm` na `rm -i` nebo `mv` na `mv -i`).
To se provede příkazy `alias cp 'cp -i'` umístěnými v souboru `~/.bashrc`, který se čte při startu terminálu. Původní příkazy jsou dostupné jako `\rm`, `\mv`, `\cp`.

Podobně > přepíše soubor **bez varování**

Náprava:

```
guest@403-a324-01:~ $ set -o noclobber
```

Chceme-li soubor přepsat, použijeme `>|` místo `>`

```
guest@403-a324-01:~ $ echo "Klotylda" >| jmena.txt
```

- Oboje je nastaveno pro uživatele `guest` na klastrech/počítačích.



- Které soubory začínají na písmena a nebo A?

```
guest@403-a324-01:~ $ touch aha.txt # timestamp; neexistuje-li soubor, založí prá  
guest@403-a324-01:~ $ ls [aA]*
```

Tato expanze je vlastností shellu a funguje nejen po `ls`.

`?` nahrazuje 1 znak

`*` nahrazuje 0 nebo více znaků

`[a-z]` nahrazuje všechna malá písmena (bez diakritiky)

`aha.txt`, `Aha.txt`, `AHA.TXT`
jsou různé soubory

je více možností, hledejte
regular expression

- Napiš cokoliv na terminál:

```
guest@403-a324-01:~ $ echo "cokoliv"
```

- Napiš na terminál v řádku soubory začínají na písmena aA:

```
guest@403-a324-01:~ $ echo [aA]*
```

Napiš na terminál `"[aA]*"`:

```
guest@403-a324-01:~ $ echo "[aA]*"
```

lze též použít jednoduché apostrofy, význam je trochu jiný

Uvozovky (dvojitě i jedno-
duché) zabraňují „expanzi“
podle nalezených souborů.

Tak tohle byl základ, který potřebujete pro zvládnutí úloh z počítačové chemie.

Další text vám neuškodí, ale není nutný.

Ovšem pokud rádi klikáte místo příkazů, podívejte se ještě na [Midnight Commander](#).

- Spočítej druhou odmocninu ze 3 na 5 desetinných míst:

```
guest@403-a324-01:~ $ bc  
sqrt(3.00000)
```

Ctrl-D

nebo

```
guest@403-a324-01:~ $ echo "sqrt(3.00000)" | bc
```

- Je Mersennovo číslo $2^{67} - 1$ prvočíslo [F. N. Cole (1903)]?

```
guest@403-a324-01:~ $ echo "2^67-1" | bc | factor
```

- Seřad' jména podle abecedy

Vytvoř soubor se jmény (na terminalu radeji bez hacku a carek):

```
guest@403-a324-01:~ $ cat > jmena.txt
```

Lionel

Izidor

Perchta

Ctrl-D

```
guest@403-a324-01:~ $ sort < jmena.txt
```

- Ještě existuje `stderr`, kam obv. jdou chyby. Dá se separovat pomocí `2>`, oba výstupy jsou `&>`

| je roura (*pipe*), která posílá výsledek jednoho příkazu (`stdout`) do dalšího jako vstup (`stdin`)

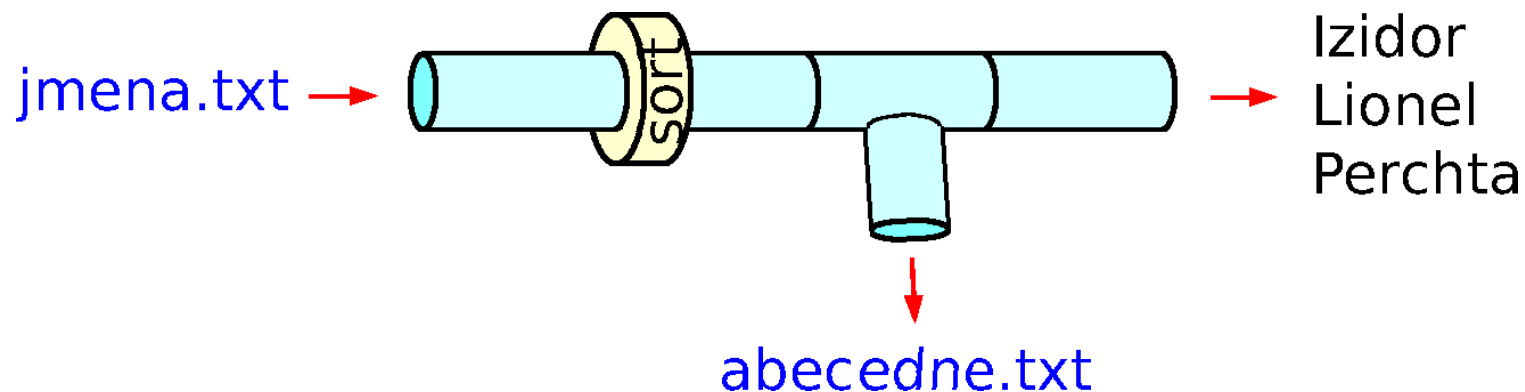
`bc` a `factor` jsou v tomto kontextu „filtry“

> pošle `stdout` do souboru
>> přidá `stdout` na konec souboru (*append*)
< značí vstup ze `stdin`

- `cat` (*concatenate*) spojuje soubory. Zkuste
`guest@403-a324-01:~ $ cat jmena.txt jmena.txt`

- Tyto dva příkazy jsou ekvivalentní:
`guest@403-a324-01:~ $ sort < jmena.txt`
`guest@403-a324-01:~ $ cat jmena.txt | sort`

- T-kus (do souboru a zároveň pošle dál):
`guest@403-a324-01:~ $ cat jmena.txt | sort | tee abecedne.txt`



- Prohlížení dlouhého textového souboru
`guest@403-a324-01:~ $ less S0UB0R`
procházení: šipky, `PgUp`, `PgDn`
hledání = `/`(tam), `?`(zpět)
quit = `q`

Proč `less`?

Protože historicky `more` (bez vracení)
existovalo v Unixu i DOSu.

- Umístění, aliasování:

```
guest@403-a324-01:~ $ type C0
guest@403-a324-01:~ $ which C0
```

- Typ souboru, verze, velikost obrázku...

```
guest@403-a324-01:~ $ file SOUBOR
```

- Stav připojených disků

```
guest@403-a324-01:~ $ df
```

- Info o procesoru, paměti

```
guest@403-a324-01:~ $ cat /proc/cpuinfo
guest@403-a324-01:~ $ cat /proc/meminfo
```

- Jakou verzi linuxu mám?

```
guest@403-a324-01:~ $ lsb_release -a
```

- Kdo tu pracuje?

```
guest@403-a324-01:~ $ w
```

- Jak dlouho a jak počítač pracuje?

```
guest@403-a324-01:~ $ uptime
```

- Tisk

```
guest@403-a324-01:~ $
                        lp [-d TISKARNA] SOUBOR
guest@403-a324-01:~ $
                        lpr [-P TISKARNA] SOUBOR
```

- Najdi soubor:

z databáze (rychlé, nemusí být aktuální):

```
guest@403-a324-01:~ $ locate SOUBOR
```

přímo z `./` (pomalé):

```
guest@403-a324-01:~ $
                        find . -name SOUBOR
```

jsou programem ve vhodném interpretovaném jazyce. Pro řízení výpočtů na linuxových klastrech jsou vhodné jazyky `bash` (jazyk, ve kterém pracuje terminál), dále `python` a jeho nadstavby.

Dobré zvyky pro `bash` skripty:

- Skript je textový soubor a má koncovku `.sh`
- První řádek je `#!/bin/bash`

Spuštění skriptu:

- `guest@403-a324-01:~ $ bash SKRIPT.sh`
- `guest@403-a324-01:~ $ chmod +x SKRIPT.sh # označí se jako spustitelný (executable)`
`guest@403-a324-01:~ $ ls -l # všimněte si -rwxrwxr-x`
`guest@403-a324-01:~ $ ./SKRIPT.sh`

Systémová proměnná `PATH` obsahuje seznam složek oddělených `:`, ve kterých se hledají spustitelné programy. Pokud obsahuje složku `.` (pracovní složka), stačí jen:

```
guest@403-a324-01:~ $ SKRIPT.sh
```

Otestovat `PATH` můžete takto:

```
guest@403-a324-01:~ $ echo $PATH
```

Napište skript pro seřazení zadaných jmen podle abecedy. Možné řešení:

```
#!/bin/bash
# serazeni seznamu jmen podle abecedy
echo "Zadej seznam jmen bez uvodnich mezer po jednom na radek a ukonci Ctrl-D:"
# pokud soubor jmena.txt jiz existuje, bude zachranen jako jmena.txt~
[ -e jmena.txt ] && mv jmena.txt jmena.txt~
cat > jmena.txt
echo -n "Pocet jmen k serazeni: "
wc -l < jmena.txt
echo "Serazena jmena:"
sort < jmena.txt
```

Zkopírujte skript do souboru `serad.sh`, např. takto:

```
guest@403-a324-01:~ $ cat > serad.sh Ctrl-Shift-V Enter Ctrl-D
```

a spusťte podle návodu na minulém slajdu. Můžete také opsat do editoru.

Abecední řazení závisí na tzv. *locale* – jazykovém nastavení. Příklady:

```
guest@403-a324-01:~ $ echo BBB > test.txt; echo aaa >> test.txt
guest@403-a324-01:~ $ LC_ALL=cs_CZ.UTF-8 sort < test.txt
guest@403-a324-01:~ $ export LC_ALL=C; sort < test.txt
```

- Proměnné v `bash`i obsahují (zpravidla) řetězce. Příklady:

```
guest@403-a324-01:~ $ X="je sunka!"
```

```
guest@403-a324-01:~ $ echo "Pocitac $HOSTNAME $X"
```

Jména proměnných rozlišují malá a velká písmena (*case sensitive*).

- Můžeme zobrazit znaky č. 4 a 5 (2 znaky od 4. znaku, čísluje se od 0):

```
guest@403-a324-01:~ $ echo ${X:3:5}
```

- Tohle se hodí ve skriptech – náhrada koncovky (% znamená konec řetězce):

```
guest@403-a324-01:~ $ IN=soubor.com
```

```
guest@403-a324-01:~ $ OUT=$IN/%.com/.out
```

```
guest@403-a324-01:~ $ echo $OUT
```

- Výsledek příkazu (tj. `stdout`) do proměnné:

```
guest@403-a324-01:~ $ pocetslov=$(wc -w < jmena.txt)
```

```
guest@403-a324-01:~ $ echo $pocetslov
```

\$ uvnitř "... " funguje

\$ uvnitř '...' zůstává \$

\\$ uvnitř "... " zůstává \$

- Symbolický link (symlink, soft link $\approx$ zástupce ve Windows) je odkaz na (absolutní nebo relativní) umístění souboru nebo složky. Symlink může vést třeba na externí disk.

Příklady:

```
guest@403-a324-01:~ $ ln -s jmena.txt zastupce.txt
guest@403-a324-01:~ $ ln -s ../.bashrc
guest@403-a324-01:~ $ ls -al
```

Soubory začínající tečkou jsou „systémové“ a zobrazí se jen pomocí `ls -a`.

Tip: mám-li stejný vstupní soubor pro výpočty, je úspornější je nekopírovat, ale symlinkovat.

- Pokud editujeme symlink, editujeme odkazovaný soubor.
- Symlink můžeme přejmenovat.
- Pokud přejmenujeme původní soubor, bude symlink odkazovat špatně.

- Hard link je druhý záznam v adresáři téhož souboru na stejném souborovém systému.

Příklady:

```
guest@403-a324-01:~ $ ln jmena.txt JMENA.TXT
guest@403-a324-01:~ $ ls -l
```

(Všimněte si dvojky u souborů `jmena.txt JMENA.TXT`.)

- Všechny hardlinky jsou ekvivalentní.
- Pokud editujeme hardlink, záleží výsledek na editoru (viz další slide).
- Hardlink (kteroukoliv kopii) můžeme přejmenovat.

Příklad použití: inkrementální zálohy.

Začátečníkům použití hardlinků nedoporučuji!

Pokud editujeme symlink a uložíme změny, je to jako bychom editovali původní soubor. Editovaný soubor je dostupný pod oběma jmény.

Pokud editujeme hardlink a uložíme, záleží výsledek na použitém editoru a/nebo jeho nastavení:

- U LibreOffice, `nano`, `vi` je to stejné jako u symlinku, a to i když požadujeme při uložení souboru vytvoření zálohy (např. `nano -B`). Tato záloha zpravidla končí znakovým `~`¹ a je to kopie původního souboru (vč. původního data vytvoření) a není hardlinkem.
- Editor `emacs` přejmenuje původní verzi (hardlinkovanou) s koncovkou `~` a uloží novou, která není hardlinkem. Původní soubor je nezměněn a je hardlinkem na backup, ne na editovanou kopii.

¹některý software umísťuje zálohy do speciální složky

Trochu složitější příklad

+ 20/50
pchlm

Pro zadaný rok narození osoby najděte rok, kdy bude/byl jak věk tak letopočet prvočíslem.

Věk zaokrouhujeme dolů. Pokud osoba nemá narozeniny přesně začátkem nebo koncem roku, je její věk před narozeninami $ROK - NAROZENÍ - 1$, po narozeninách $ROK - NAROZENÍ$.

```
#!/bin/bash
if [ $# -lt 1 ]; then
    cat <<EOF
Hledej první rok, kdy jak vek tak rok jsou prvocisly.
Použití:
    prime.sh ROK
Příklad:
    prime.sh 1958
EOF
    exit 1
fi

ROK=$1

function prime ()
{
    NW=$(factor $1 | wc -w) # pocet slov v rozkladu na prvocinitele
    [ $NW == 2 ] && return 0 || return 1 # 0=true, 1=false
}
```

```
while true
do
    ROK=$(echo $ROK+1 | bc)
    prime $ROK || continue
    VEK=$(echo $ROK-$1 | bc)
    prime $VEK && break
    VEK=$(echo $ROK-$1-1 | bc)
    prime $VEK && break
done

echo "*= $1 ROK=$ROK VEK=$VEK"
```

Deklarujeme-li `declare -i ROK`, lze napsat jen `ROK=$ROK+1` (jen pro celá čísla).

je nadstavba shellu podobná aplikaci Total Commander resp. Windows Commander. Je vhodná pro uživatele zvyklé na Windows.

- nastartujte Midnight Commander příkazem

```
guest@403-a324-01:~ $ mc
```

- Na účtu `guest` může být obrazovka z důvodu činnosti ostatních v nestandardní pozici. Pak pomocí `Tab` přejděte na panel, který má nahoře vlnovku (~)². Základní ovládání:

<code>Ctrl-O</code>	zobrazit výpisy na obrazovku/commander (přepínač)
<code>Ctrl-L</code>	spravit rozbitou obrazovku (po výstupu)
<code>F3</code>	prohlížení souboru (předefinováno pro některé typy)
<code>F4</code>	editace textového souboru
<code>Shift-F4</code>	nový textový soubor + editace
<code>F9</code>	menu
<code>Enter</code> , doubleclick	akce (start asociované aplikace, změna složky)
<code>F10</code>	ukončit Midnight Commander

- Soubory jsou asociovány s aplikacemi, a to i soubory simulačního balíku MACSIMUS

²Vlnovka značí domovskou složku uživatele, zde ~ = /home/guest

- V některých distribucích je `F10` zablokováno funkcí menu terminálu. Pak:
 - použijte `Esc` `0` (po sobě)
 - použijte příkaz `exit`
 - změňte přiřazení `F10` v nastavení terminálu
- V Midnight Commanderu nefunguje posuvník okna
- Jste-li v Midnight Commanderu a na obrazovce je smetí, stiskněte `Ctrl-L`
- Nevíte-li, kde jste, pak `Ctrl-O` `Ctrl-O`
- Pokud jste omylem stiskli `Ctrl-S` (stop výstupu na terminál), napravíte to pomocí `Ctrl-Q`
- Někdy pomůže `Ctrl-C` = přerušení
- **V nouzi** `Ctrl-Z`, příkaz `jobs` a pak `kill %1` atd. podle počtu jobů

Od Win 10 je dostupný “Windows Subsystem for Linux”, který si instalujete [podle návodu pro Win 11 a novější 10](#), případně [WSL pro Win 10](#). Jako distribuci doporučuji [Ubuntu](#). Zvolte si shell [bash](#). Po otevření terminálu si aktualizujte systém a doinstalujte chybějící software takto (příkaz [cd](#) nastaví váš domovský linuxový adresář, [sudo](#) bude chtít heslo, které jste si zvolili při instalaci):

```
cd
sudo apt update && sudo apt upgrade
sudo apt autoremove
sudo apt install gcc libx11-dev libncurses-dev libncurses5
sudo apt install x11-apps
```

Chce-li systém doinstalovat další balíčky, potvrďte.

Vhodné změny v souboru [.bashrc](#) v domovském adresáři (pomocí příkazu [nano .bashrc](#) přidáte na konec, za <#> je komentář):

```
alias rm='rm -i' # ptá se před smazáním souboru
alias cp='cp -i' # ptá se před přepsáním souboru
alias mv='mv -i' # ptá se před přepsáním souboru
PATH="/home/$USER/macsimus/bin:$PATH" # přidá, kde hledat spustitelné soubory
export BLENDPATH="/home/$USER/macsimus/blend/data" # soubory s parametry
```

Poprvé načtete příkazem [source .bashrc](#), příště se načte samo.

Pro přesunování mezi souborovým systémem Windows doporučuji z vašeho linuxového domovského adresáře udělat link („zástupce“), například (disk C: → /mnt/c):

```
cd
ln -s /mnt/c/Users/Plocha
```

Nyní vidíte obsah vaší plochy jako složku **Plocha**. (Např.) pomocí Midnight Commanderu můžete přesouvat soubory. Ale silně nedoporučuji pracovat přímo v této složce!

Nyní stáhnete (ale nerozbalíte/neextrahujete) ve Windows z webu ZIP s **MACSIMUS**em na plochu. Z linuxu provedete (příklad):

```
unzip Plocha/macsimus-2022-10-27.zip
cd macsimus
./install.sh gcc
```

Více o instalaci

Nyní zkuste napsat:

```
plot "[99:0:7]:sin(x):cos(x)"
```

Pokud se vám zobrazí elipsa, je vše v pořádku.

Pokud se hlásí něco jako “**Command 'plot' not found**”, máte špatně **PATH** nebo neinstalované **libx11-dev** apod.

Pokud se hlásí něco jako “**file '[99:0:7]:sin(x):cos(x)' not found**”, patrně máte špatně **PATH** a našel se jiný plot. Zkuste `/home/$USER/macsimus/bin/plot` a/nebo opravte **PATH**.

Pokud se hlásí něco jako “**Can't open display:**” nebo “**XDisplay not found**” (tj. máte WSL bez grafiky), zkuste:

- `export DISPLAY=:0`

- Z PowerShellu přinstalovat: `wsl --update`

- Na starších Windows 10 nebo v případě přetrvávajících potíží zkuste externí X11 server **Xming**. Instalujte a případné dotazy potvrďte, spusťte. Ve stavovém řádku se musí objevit ikona .

Pro připojení ze vzdáleného počítače (např. školních Windows) k serveru klastru musíte mít:

- Terminál s příkazovým řádkem, na kterém běží interpret příkazů zvaný *shell*, typicky **bash**.
Pro Windows lze použít **MobaXterm** nebo **PuTTY**.
Pro macOS viz zde.
- Volitelně: „grafická“ nadstavba terminálu podobající se otevřené složce v GUI, např. **Midnight Commander**. Na klastrech je instalován, dostupná je asociace typů souborů pro MACSIMUS.
Spustí se takto:
`guest@403-a324-01:~ $ mc`
vyskočí se z něj stiskem **F10**.
- **X server** pro zobrazení grafiky. Je součástí MobaXtermu, pro připojení PuTTY lze použít **Xming**.
Příklad (zobraz hodiny):
`guest@403-a324-01:~ $ xclock`

Grafika se počítá na vzdáleném počítači (klient), na vaši obrazovku (server) se pošle okno k zobrazení.

Pokud budete sedět ve škole, můžete tuto stránku přeskočit

● VPN do školy + připojit se na „virtuální učebnu“ (doporučuji “Classroom / Study EN”):

● Webové rozhraní:

<https://vdistudovna.vscht.cz/portal/webclient/index.html>
(otestováno a doporučeno)

● Aplikace: <https://vc.vscht.cz/software/virtualni-ucebna>
(možná vám bude fungovat, ale já jsem měl problémy)

Po určité době nečinnosti vyžaduje okno znovupřihlášení pomocí **Ctrl-Alt-Del**, to získáte vysunutím „pacičky“ zleva a stiskem tlačítka nahoře se třemi „okýnky“.

● Na klastr se lze připojit i přímo (bez protivného mezikroku virtuální učebny) z vašeho počítače jen pomocí VPN, ale to vyžaduje speciální povolení Výpočetního střediska. Budete-li mít zájem, ozvěte se včas, vyřízení trvá několik dní.

Dále postupujeme jako z lokálního VŠCHT počítače (viz další stránky).

Poznámky:

● Máte-li kvalitní internet, je připojení snesitelně rychlé vč. grafiky.

● Pokud je zobrazení molekul (program [show](#), viz [dále](#)) pomalé, můžete změnit zobrazení na „drátové“ stiskem **&** nebo v menu **7** nebo při startu takto:
`guest@403-a324-01:~ $ show -I '&'` další parametry

PC klastry (dávkové spouštění delších výpočtů):

● [403-a324-01.vscht.cz](#) (Argon)

● [403-as67-01.vscht.cz](#) (Neon)

Počítače (přímý výpočet – po domluvě):

● [403-a325-05.vscht.cz](#) (4 procesory po 2 vláknech)

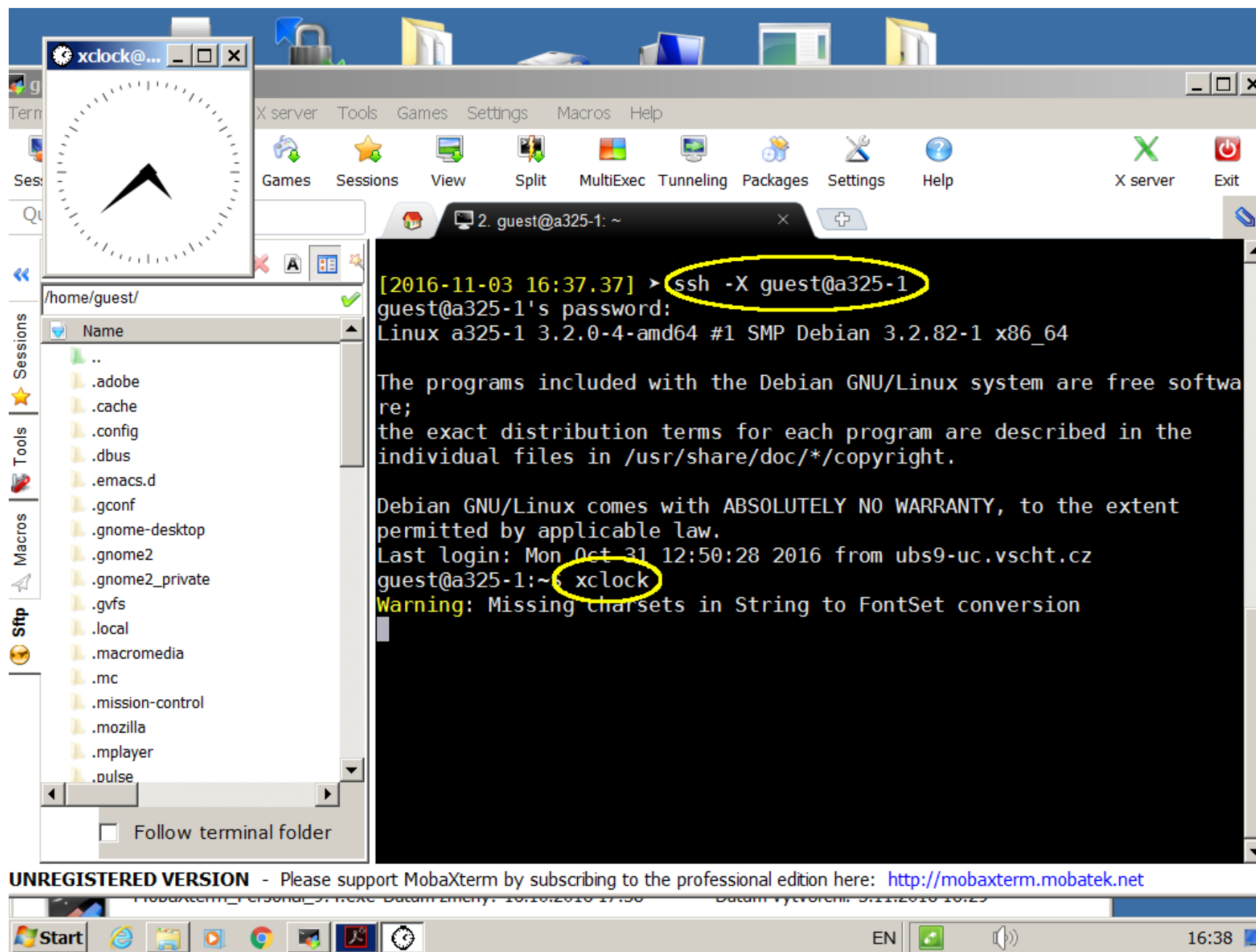
● [403-A325-16.vscht.cz](#) (Greta, 24 procesorů po 2 vláknech)

- + Terminál + grafika (X-server) v jednom
- Nezvládne zobrazit několik grafů najednou (např. pro mnoho lidí v učebně)
- Na disku “scratch” (S:) najděte složku /pocitacova_chemie/Connect/ a spusťte `MobaXterm_Personal_22.1.exe`
Nebo stáhněte [MobaXterm Home Edition \(Portable\)](#)
- Rozbalte, spusťte, potvrďte vše
- Klikněte na `+ Start local terminal`
- V okně terminálu spusťte vybranou relaci, např. (Argon):
`[2021-11-11 11:11.11] ssh -X guest@403-a324-01.vscht.cz`
Heslo vám sdělím. Během psaní hesla se nic nezobrazuje!
Jako test spusťte hodiny: `guest@403-a324-01:~ $ xclock`
Pokud se nezobrazí, mohou být ikonizované – hledejte dole na liště.
- Alternativně (v některých verzích MobaXtermu) se jméno počítače (`403-a324-01.vscht.cz`) a uživatele (guest) napíše do dialogu.
- Viz též dále metoda 2 = PuTTY + Xming (je instalované v některých počítačových učebnách)

pokud máte vlastní účet, použijte ho

Připojení na vzdálený počítač metoda 1 – MobaXterm

30/50
pchlm



Připojení na vzdálený počítač metoda 2: PuTTY + Xming

31/50
pchlm

Spolehlivější než MobaXterm, ale 2 kroky

Terminál s příkazovým řádkem (PuTTY)

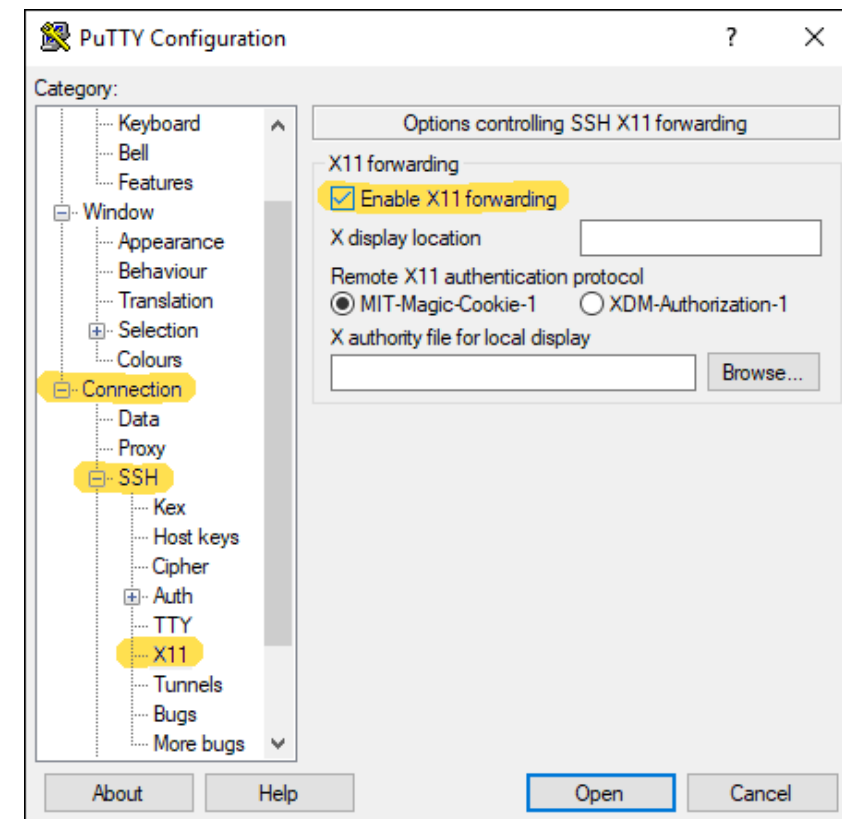
- Windows Start → hledat → `putty`, spustit
nebo `S:pocitacova_chemie/Connect/putty64bit.exe`
- Host name → `403-a324-01.vscht.cz` (nebo jiný počítač)
- Connection → **+** SSH [→ Tunnels] → X11
→ **v** Enable X11 forwarding (**nutné pro zobrazení grafiky**)
- zpět Session → Open
- Login as: guest
- Password: (sdělím)

pokud máte vlastní
účet, **použijte ho**

X server pro zobrazení grafiky (Xming)

- Windows Start → hledat → `xming` a spust'te
nebo `S:pocitacova_chemie/Connect/XLaunch.exe - Shortcut.lnk`
případně `S:pocitacova_chemie/Connect/Xming-6-9-0-31-setup.exe` a instalovat (např. na
plochu)
- Dotazy potvrdit **další**, firewall zamítnout. Ve stavovém řádku se musí objevit ikona .

PuTTY a Xming jsou instalovány
v některých počítačových učebnách



Budu předpokládat, že používáte výchozí shell `bash`. Jsou dvě možnosti:

- Použiji instalaci, která je pod uživatelem `guest`.

Nastavení prostředí (lze umístit např. do vašeho `.bashrc` nebo `.profile`):

```
guest@403-a324-01:~ $ source /home/guest/env.sh
```

příkaz `source` (nebo jen `.`) načte soubor a nezapomene nastavení.

- Instaluji si **MACSIMUS** sám/sama. Stáhnete ZIP a rozbalíte:

```
unzip macsimus-2022-10-27.zip
```

```
cd macsimus
```

```
./install.sh gcc
```

Více o instalaci

Je součástí MobaxTermu. Pokud by nefungoval, doporučuji “WinSCP”. Pokud není ve škole instalovaný, hledejte v Googlu, zvolte “Portable executables”, které nevyžadují instalaci.

Jste-li ve virtuální učebně, ukládejte na svůj disk (Z:) nebo na scratch (S:)

Možná budete dále zpracovávat soubory následujících typů:

- .g = radiální distribuční funkce
- .cpa = ASCII image konvergenčního profilu
- .z = hustotní profil (ve směru z)

Všechny tyto soubory jsou textové s tím, že konce řádků jsou ukončeny LF, zatímco pod Windows se ukončují dvojicí CR LF. Starší Windows nemusí tyto soubory správně zobrazit. Pak pro konverzi do Windows použijte příkaz (na klastru):

```
guest@403-a324-01:~ $ unix2dos JMENO_SOUBORU
```

iOS ukončuje řádky CR

V návodu máte vstupní soubor. Jak ho zkopírovat na klastr?

Ve Windows označíme daný text a stiskneme `Ctrl-C` jako obvykle.

V okně terminálu:

- `Ctrl-Shift-V`, to ale nefunguje uvnitř Midnight Commanderu.
- Pravé nebo prostřední tlačítko myši (podle nastavení prostředí).
- Pokud jsou problémy uvnitř editoru, lze pomocí utility `cat` z příkazového řádku takto:

```
guest@403-a324-01:~ $ cat > S0UB0R
```

`Ctrl-Shift-V` nebo prostřední tlačítko myši (příp. pravé)

Není-li kurzor na začátku řádku, `Enter` `Ctrl-D`

Pokud soubor již existuje, odmítne se přepsat. Pak lze provést:

```
guest@403-a324-01:~ $ cat >| S0UB0R
```

- Máte-li vlastní počítač s linuxem, WSL, MinGW aj. nebo vlastní účet na klastru:
 - stáhněte si instalační archiv z [GitHubu](#) nebo ze [školního serveru](#)
 - rozbalte na vhodné místo (ZIP archiv obsahuje cestu `macsimus/`)
 - spusťte instalační skript:
`./install.sh gcc`
 - Důležitou součástí nastavení je instalace prostředí. Proveďte se přidáním následujících příkazů do souboru `.bashrc`:
`export BLENDPATH=/home/$USER/macsimus/blend/data`
`export PATH=/home/$USER/macsimus/bin:$PATH`
a (poprvé) provedením příkazu
`source .bashrc`
- Jste-li přihlášení jako guest na školním počítači s linuxem, neinstalujte a nenastavujte, ale musíte pracovat ve vaší složce (v tomto návodu označeno jako JOSEFK). Tu vytvoříte příkazem
`guest@403-a324-01:~ $ mkdir JOSEFK`
`guest@403-a324-01:~ $ cd JOSEFK`

Typický příkaz, kterým se spouští simulace, je

```
guest@403-a324-01:~/sim $ cookewslc ff sim
```

- `ff.ble` je silové pole (vyrobené programem `blend` či získané jiným způsobem).
- Druhý parametr `sim` se vztahuje ke dvěma vstupním souborům:
 - `sim.def` (chybí-li, hledá se `ff.def`). Typicky obsahuje počet molekul a další společné parametry. Soubor končí středníkem.
 - `sim.get`. Řídicí soubor typicky obsahuje počet cyklů `no`, parametr `init` a další parametry. Po každém středníku se čtou další data a simulace se restartuje. Výsledné soubory jsou tvaru `sim.*`.

Tyto soubory budete mít pro každou úlohu připraveny i s popisem použitých proměnných.

- Parametr `init` (výběr možností)
 - `init="random"` Nastřílí molekuly náhodně
 - `init="slab"` Vytvoří vrstvu (náhodně)
 - `init="crystal"` Molekuly v pravidelné mřížce
 - `init="append"` Přečte uložená `sim.cfg` a pokračuje, trajektorie a konvergenční profily se prodlouží, měření se restartuje
 - `init="start"` Přečte uložená `sim.cfg` a restartuje trajektorie, konvergenční profily i měření

Pokud simulace (`cook*`) zhavaruje, může se tak stát dvojím způsobem:

- Úplně spadne, pak zůstane ve složce soubor `SIMNAME.loc`. (`SIMNAME` je poslední argument `cook ... SIMNAME`). Smažte ho, ale jen pokud jste si jisti, že simulace skutečně neběží! Dvojí start simulace se stejným `SIMNAME` = problémy.
- Ohlásí chybu, tu najdete jednak na konci souboru `SIMNAME.prt`, na klastru také v souboru `JMENO_JOB.o123456` (kde 123456 je číslo jobu), v případě přímého spouštění i na obrazovce.

Přerušení běžící simulace (soubory se uzavřou):

`Ctrl-C` pokud běží přímo

`guest@403-a324-01:~/sim $ touch SIMNAME.stp` pokud běží v pozadí nebo na jiném počítači
klastru – může chvíli trvat.

Textové soubory jsou složeny z písmen a řídicích znaků jako je CR a LF. Ve Windows se editují zpravidla aplikací Notepad (Poznámkový blok) a mají koncovku .txt, která často není vidět³. Pod linuxem je nutno vždy psát koncovku souboru!

- Úpravy textových souborů – Midnight Commander:

Úpravy stiskem **F4**

Nový soubor **Shift-F4**

Řádek se ukončuje:

linux: LF

Windows: CR LF

iOS: CR

- Dva jednoduché editory dostupné z příkazového řádku a běžící v terminálu:

`guest@403-a324-01:~/sim $ nano S0UB0R # doporučený pro začátečníky`

`guest@403-a324-01:~/sim $ mcedit S0UB0R # stejný editor jako v Midnight Commanderu`

Pozn.: ^ v návodu značí **Ctrl-**, M- značí **Alt-**

Clipboard se ovládá v okně terminálu pomocí **Ctrl-Shift-C** a **Ctrl-Shift-V**. Ve většině editorů (ne v `mcedit`) funguje (jiný) clipboard pomocí prostředního nebo pravého tlačítka myši (podle nastavení), z okna do okna lze pak snadno kopírovat označený text bez doteku klávesnice.

Soubory `.def`, `.get` (řídicí soubor simulace), `.cpi`, `.che`, `.ble`, `.prt` (výstupní protokol) v MACSIMUSs jsou textové a lze je editovat.

³Silně doporučuji odškrtnout volbu „skrýt koncovku souborů známých typů“ v Průzkumníku Windows.

blend je potřeba k úloze „Vibrace plátek grafenu“.

- kliknutí pravým tlačítkem myši na tlačítko dá návod pro tlačítko, zrušte druhým kliknutím
- myš posun (drag):
levá = točit, prostřední (tj. kolečko) = posun, pravá = točit/velikost
- myš click:
levá = označit, prostřední = celá molekula po vazbách, pravá = odznačit
- myš kolečko: zoom
- menu on/off: ≡ nebo F10

tlačítko	klávesnice	funkce
grid	=	mřížka (po 1 Å)
move	m	hýbe se celou molekulou / označenou částí
CG	,	optimalizace (Conjugate Gradients)
rand	:	randomizace + optimalizace
finish	.	ulož a skonči (resp. pokračuj další molekulou)

Návod pro show:

- kliknutí pravým tlačítkem myši na  dá návod pro tlačítko, zrušte druhým kliknutím
- kliknutí označuje molekuly (asi nebudete potřebovat)
- tažení rotuje a pohybuje konfigurací:
 - levé tlačítko: rotace okolo $\hat{x}$, $\hat{y}$
 - prostřední tlačítko: přesun
 - pravé tlačítko: rotace okolo $\hat{z}$
- kolečko myši = zoom
- menu on/off:  nebo 
- Start trajektorie: 
- Z `show` získáte “printscreen” (kromě PrintScreen) stiskem  nebo kláves + (lze získat i sérii pro případnou výrobu videa či animovaného GIFu⁴). Výsledek je ve formátu “portable pixel map” (raw .ppm, verze P6). Tento soubor lze načíst z Windows např. programem IrfanView, nebo přímo na místě převést pomocí [pnmtopng](#), [pnmtjpg](#).

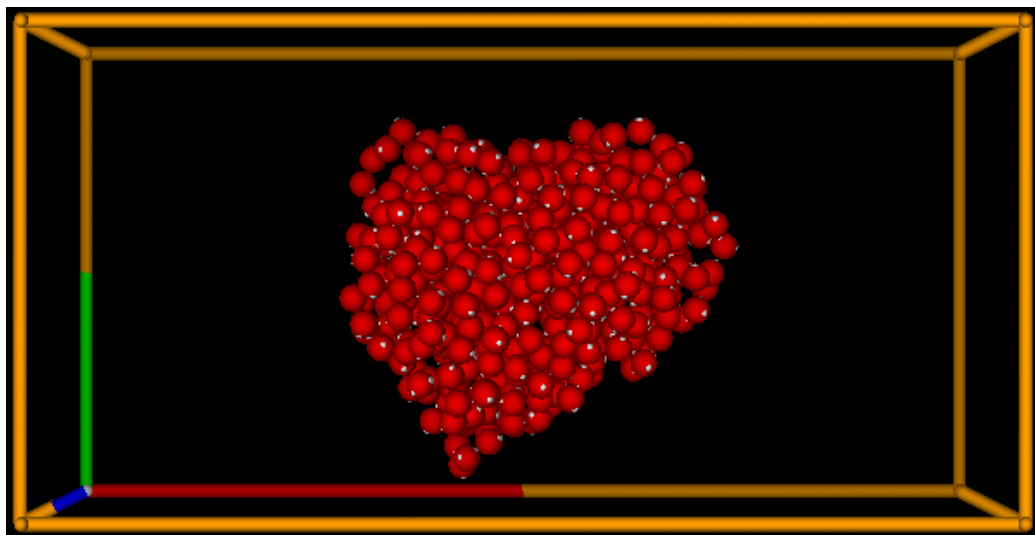
⁴vyžaduje utilitu [convert](#) z ImageMagick

Pěkné obrázky získáte pomocí techniky “ray tracing”, implementován je `ray`, Mark VandeWettering “reasonably intelligent raytracer”. Z `show` získáte popis scény ve formátu `nff` stiskem `NFF` nebo kláves `[N]+[o]` – pouze v nedráťových režimech (prvních 5). Pak buď zvolte `one frame+render` (`[O]`), nebo jen `one frame` (`[o]`), pak se renderuje z `mc` nebo příkazem:

```
guest@403-a324-01:~/sim $ ray -n mojevoda.0000
```

Opět se vytvoří obrázek ve formátu `.ppm`.

Bílé pozadí získáte startem `show -bgFFFFFF` nebo změnou příkazu `b 1 1 1` na `b 1 1 1` v souboru `mojevoda.0000.nff`.



K exportu animovaného GIFu musíte mít instalovaný ImageMagick. Pokud není a máte přístup administrátora, na Ubuntu a pod. se instaluje takto:

```
guest@403-a324-01:~/sim $ sudo apt install imagemagick
```

K exportu stisknete **PIC** → **animated GIF**. Všechny obrazovky se nyní zapisují (ve formátu **.ppm** = Portable Pixel Map, P6). Chcete-li zaznamenat rotaci, než myš je lepší používat hot keys **x**, **X**, **y**, **Y** a úhel nastavit pomocí **/**, *****. Po stisku **End series** se vygenerovaná série **ppm** souborů zkonvertuje na animovaný GIF. Pro změnu parametrů pro ImageMagick použijte **show** option **-L** (viz manuál pro ImageMagick).

Doporučuji skript `jsub`

- Spustíte simulaci příkazem (příklad)

```
guest@403-a324-01:~/sim $ jsub -n VASEJMENO cook nacl NPT
```

`jsub` = skript nastavující prostředí a požadující spuštění na uzlu klastru

`-n VASEJMENO` jméno úlohy (zobrazuje se max. 10 znaků)

`cook` simulační program

`nacl` první parametr pro `cook`: použije se `nacl.ble` a `nacl.def`

`NPT` druhý parametr pro `cook`: jméno simulace (vstupní data = `NPT.get`)

- Kontrola jobu

```
guest@403-a324-01:~/sim $ jstat
```

- Zrušení jobu

```
guest@403-a324-01:~/sim $ qdel CISL0J0BU
```

- Přerušení správně běžícího výpočtu `cook`:

Midnight Commander: doubleclick `NPT.loc`

```
guest@403-a324-01:~/sim $ touch NPT.stp
```

- Paralelní výpočet – příklad (vyžaduje paralelní verzi, např. `cookewslcP1`):

```
guest@403-a324-01:~/sim $ jsub -n MYJOB -q aq -p 3 cookP1 nacl NPT
```

kde `aq` = fronta (tato je na Argonu), `3` = počet vláken (běžně max. 4 pro malé úlohy)

- Spustíte simulaci příkazem:
`guest@403-a324-01:~/sim $ cook nacl NPT`
`cook` simulační program
`nacl` první parametr pro `cook`: použije se `nacl.ble` a `nacl.def`
`NPT` druhý parametr pro `cook`: jméno simulace (vstupní data = `NPT.get`)
- Můžete spustit více běhů paralelně (podle počtu jader vašeho procesoru). Bud' přerušíte aktivní výpočet pomocí `Ctrl-Z` a pak příkazem `bg` dáte do pozadí, nebo rovnou zadáte jako
`guest@403-a324-01:~/sim $ cook nacl NPT &`
Můžete ovšem také nastartovat víc terminálů.
- Přerušení správně běžícího výpočtu `cook`:
Midnight Commander: doubleclick `NPT.loc`
`guest@403-a324-01:~/sim $ touch NPT.stp`

Součástí MACSIMUSu – jednoduché kreslení grafů.

Návod dostanete spuštěním bez parametru. Příklad:

```
guest@403-a324-01:~/sim $ plot drop.cpa:0:1
```

levá myš vybrat obdélník = zoom in, kolečko = zoom Y

`undo` = `u` = zpět zoom, `init` = `k` = plná velikost

`PrtScr` = `@+M` = screenshot (opačné barvy, ve formátu ppm)

Lze i výstup ve formátu EPS: založte soubor `ps.def` podle příkladu:

```
s 12          # velikost fontu v pt
w 10cm 10cm   # velikost grafu (bez os) v cm
x 1cm 2mm POPIS-OSY-X (recky=\ index/exponent = _^ )
r 90 # rotace
y 1cm 2mm POPIS-OSY-Y
l X Y TEXT
l X +DY $2-POPIS-CARY2
t 1 1 # tloustky car/ramecku v pt (1pt=0.35 mm)
f 2 2 5 # hustota popisu osy x, y, pacicky v pt
```

Z okna `plot` klikněte `EPS` nebo hotkey `#` nebo `F4`: vytvoří `plot.eps`

Další možnosti v linuxu:

- gnuplot
- xmgrace
- python (library matplotlib)
- MatLab
- Maple
- LibreOffice calc

K dispozici jsou následující verze MD programu, není-li řečeno jinak, všechny obsahují Lennard-Jonesův potenciál

cookewslc – verze s elektrostatickými interakcemi počítanými tzv. Ewaldovou sumací, tj. sčítáním přes „všechny“ periodické obrazy do nekonečna (matematickým trikem). Použita je “domain decomposition” simulační buňky na malé kvádríky, metoda je vhodná pro středně velké až velké systémy. Přidán je kód pro tvorbu “slab geometry” vč. povrchového napětí, hustotních profilů aj. Totožné s **cook** ze cvičení „zonální tavba“.

cookewslcP1 – jako výše, paralelní verze, lze spustit na několika (obv. 2–8) procesorech, pro vaše úlohy asi max. 2 nebo 3

cookceslc – jako **cookewslc** ale elektrostatika je ošizená – useknutá (a vyhlazená). Mnohem rychlejší, ale pro některé úlohy nevhodné.

cookceslcP1 – jako výše, paralelní verze

cookfree – verze s vakuovými (volnými) okrajovými podmínkami

cookstars – verze s WCALJ potenciálem, vakuovými okrajovými podmínkami a gravitací místo elektrostatiky (stejně náboje se přitahují)

Platí pro nativní linux i WSL. Počet procesorů zjistíte příkazem:

```
guest@403-a324-01:~/sim $ cat /proc/cpuinfo
```

Místo příkazu `cookewslc` pak můžete použít (příklad)

```
guest@403-a324-01:~/sim $ NSLOTS=3 cookewslcP1 ff sim
```

Na klastru zadáváte úlohu do fronty. Pomocí skriptu `jsub` (zjednodušuje systémový skript `qsub` pro určité typy úloh) zadáte počet požadovaných vláken pomocí parametru `-p`:

```
guest@403-a324-01:~/sim $ jsub -p 3 cookewslcP1 ff sim
```

odhlášení	exit
přehled nedávno zadaných příkazů	history
změna složky (adresáře)	cd SLOŽKA
— zpět	cd ..
výpis souborů ve složce	ls
— podrobně některých	ls -l a*.g
výpis obsahu (krátkého ASCII) souboru	cat SOUBOR
smazání souboru	rm SOUBOR
kopírování souboru (KAM=soubor n. složka)	cp SOUBOR KAM
přesun či přejmenování souboru	mv SOUBOR KAM
editace (nového nebo starého) souboru	mcedit SOUBOR, nano SOUBOR
přerušení běžícího programu	Ctrl-C

- Nevidíte-li prompt, protože ho překryl text, stiskněte **Enter**
(Midnight Commander: **Ctrl-O** **Ctrl-O**)
- **Ctrl-C** v terminálu není „Copy“, Copy-Paste je **Ctrl-Shift-C**, **Ctrl-Shift-V**
- Text v terminálu se po označení myši kopíruje prostředním (MobaXterm) nebo pravým (PuTTY) tlačítkem myši (nastavení lze změnit)

- Asociovaná aplikace se spustí z Midnight Commanderu (mc) dvojklikem nebo Enter.
Prohlížení souboru (smysl má pro ASCII) je F3 s výjimkou .rdf
- Z příkazového řádku příkazem start, další pak starts.

typ	obsah	aplikace	Midnight Commander akce
.che	chem. vzorec	blend	editace, optimalizace s použitím silového pole, vibrační módy
.mol	mol. topologie	blend	editace, optimalizace s použitím silového pole
.plb	trajektorie	show	prohlížeč trajektorie
.cp	konvergenční profil	showcp+plot	zobrazí konvergenční profily
.cfg	konfigurace	showcfg+plot	zobrazí konfiguraci
.sta	naměřená data	staprt	statistická analýza výsledků (F3 =podrobně)
.rdf	párový histogram	rdfg+plot	zobrazí radiální distribuční funkce F3 = kumulativní distr. f. (koordinační číslo)
.g	RDF	plot	zobrazí radiální distribuční funkci
.cn	kumulativní RDF	plot	zobrazí (kumulativní) distribuční funkci
.def	parametry simulace	go	provede příkaz v 1. řádku souboru
.get	řízení simulace	go	provede příkaz v 1. řádku souboru
.loc	lock-file	stop.sh	přeruší simulaci (bez ztráty dat)

Méně používané asociace:

typ	obsah	aplikace	Midnight Commander akce
.nff	data scény	ray	raytracer vyrenderuje a zobrazí scénu
.zbuf	z-buffer	stereo	stereogram
.cpz	komprimovaný .cp	showcp+plot	zobrazí konvergenční profily
.cpa	ASCII obraz .cp	showcpa+plot	zobrazí vybrané sloupce
.atm	molekula	showatm+show	zobrazí

Formát .atm (vhodný např. pro Gaussian) je:

počet_atomů

prázdný řádek (příp. velikost boxu x y z)

Atom x y z

Atom x y z

...

Atom x y z

kde Atom = značka prvku nebo jeho atomové číslo a údaje jsou v Å