



# Přednáška 3

Systemy souborů OS UNIX.

Nástroje pro práci se souborovým systémem.

Katedra počítačových systémů FIT, České vysoké učení technické v Praze

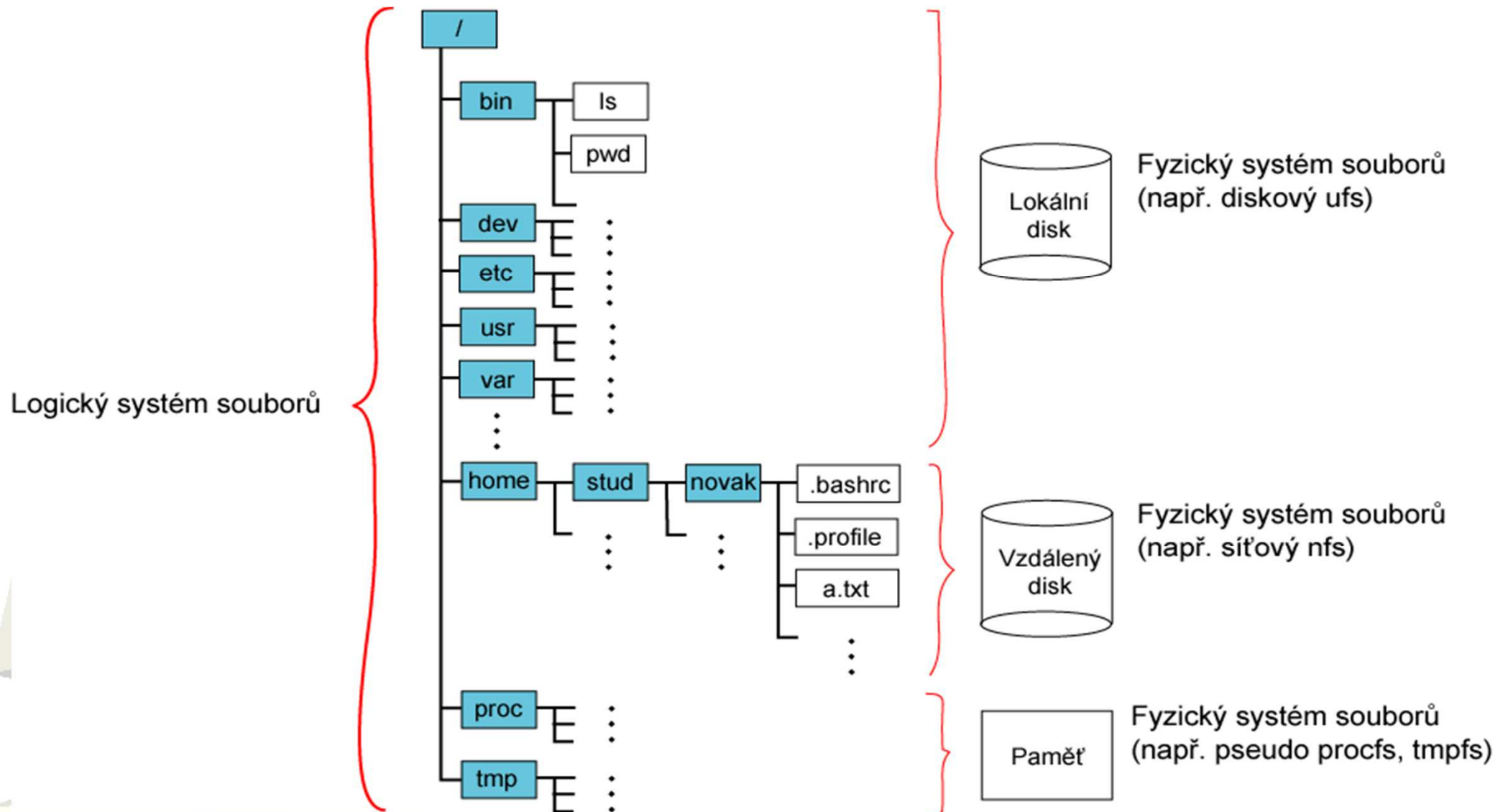
©Jan Trdlička, 2011

*Příprava studijního programu Informatika je podporována projektem financovaným z  
Evropského sociálního fondu a rozpočtu hlavního města Prahy.*

*Praha & EU: Investujeme do vaší budoucnosti*



# System souborů (FS)





- **Logický systém souborů**

- pro běžného uživatele se jeví jako jediná homogenní struktura (strom adresářů)
- přistupujeme k němu např. pomocí příkazů `cd`, `pwd`, `ls`, `cp`,...
- ve skutečnosti je tvořen jedním nebo více fyzickými systémy souborů (viz příkazy `mount`, `df`)

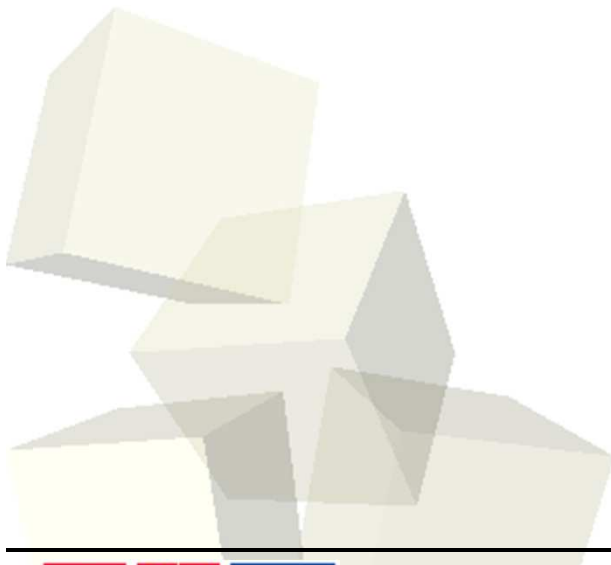
- **Fyzický systém souborů**

- podstrom adresářů, které je celý uložený na jednom fyzickém mediu (lokální disk, vzdálený disk, paměť,...)
- diskový FS (ext2, ext3, ufs, vxfs, ...)
- síťový FS (nfs,...)
- pseudo FS (profs, tmpfs, fdfs,...)



- **Výhody této koncepce**

- za pokrytí logického FS fyzickými FS je odpovědný administrátor
- pokrytí lze měnit bez vlivu na logický FS
- možnost zvětšování kapacity logického FS bez změny jeho struktury
- ohraničení chyb





# Některé důležité adresáře

/bin	Spustitelné programy
/sbin	Programy pro start a administraci systému
/dev	Speciální soubory
/etc	Konfigurační soubory
/home	Domovské adresáře uživatelů
/lib	Knihovny
/tmp	Dočasné soubory (mohou být smazány při restartování systému)
/opt	Aplikace, které nejsou součástí systému
/usr	Systémové soubory, které nemění svou velikost
/var	Systémové soubory, které mění svou velikost



**Soubor = jméno (jména) + atributy + data**

## • Jméno souboru

- maximální délka závisí na implementaci (obvykle 255 znaků)
- kódování závisí na implementaci (ASCII, UTF8,...)
- jsou povoleny všechny znaky kromě lomítka (/)
- malá a velká písmena se rozlišují
- **jména začínající tečkou** (skryté soubory/adresáře) se:
  - nenahrazují na příkazové řádce při použití znaků \* a ?
  - příkaz **ls** je nevypisuje (lze ale vynutit přepínačem **-a**)
- **jména tečka (.) a dvě tečky (..)** jsou rezervována pro
  - — aktuální adresář
  - — nadřazený adresář



- **Atributy souboru** (lze vypsat např. příkazem `ls -l`)
  - typ souboru:

<code>d</code> (directory)	adresář
<code>-</code>	obyčejný soubor
<code>c</code> (character device)	speciální znakový soubor
<code>b</code> (block device)	speciální blokový soubor
<code>l</code> (symbolic link)	symbolický link
<code>p</code> (named pipe)	pojmenovaná roura

- vlastníci souboru (uživatel, skupina)
- přístupová práva (r – čtení, w – zápis, x – spuštění, setXid, ACL,...)
- čas (vytvoření, modifikace, přístupu)

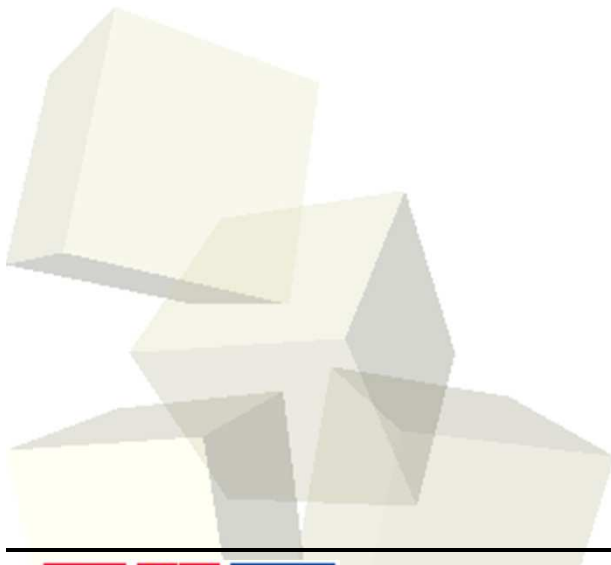


- **Data**

- obsah souboru je uložen v datových blocích na disku

- **Přístup k souboru**

- pomocí systémových volání: `open()`, `close()`, `seek()`, `read()`, `write()`, `stat()`, ...
  - příkazy OS: `more`, `less`, `cp`, `rm`, `mv`, `ln`, ...







- Umožňují **hierarchické uložení informací** ve stromě adresářů.
- **Absolutní (úplná) cesta**
  - začíná vždy v kořenovém adresáři `/`
  - obsahuje posloupnost všech adresářů (oddělených znakem `/`) mezi `/` a cílovým souborem

`/home/rocnik1/skupina12/Novak`

- **Pracovní (aktuální) adresář**
  - lze zobrazit příkazem `pwd`
  - jeho hodnota je uložena v proměnné `PWD`
  - mění se příkazem `cd nový-pracovní-adresář`
  - je určen absolutní cestou
  - každý proces může mít jiný pracovní adresář



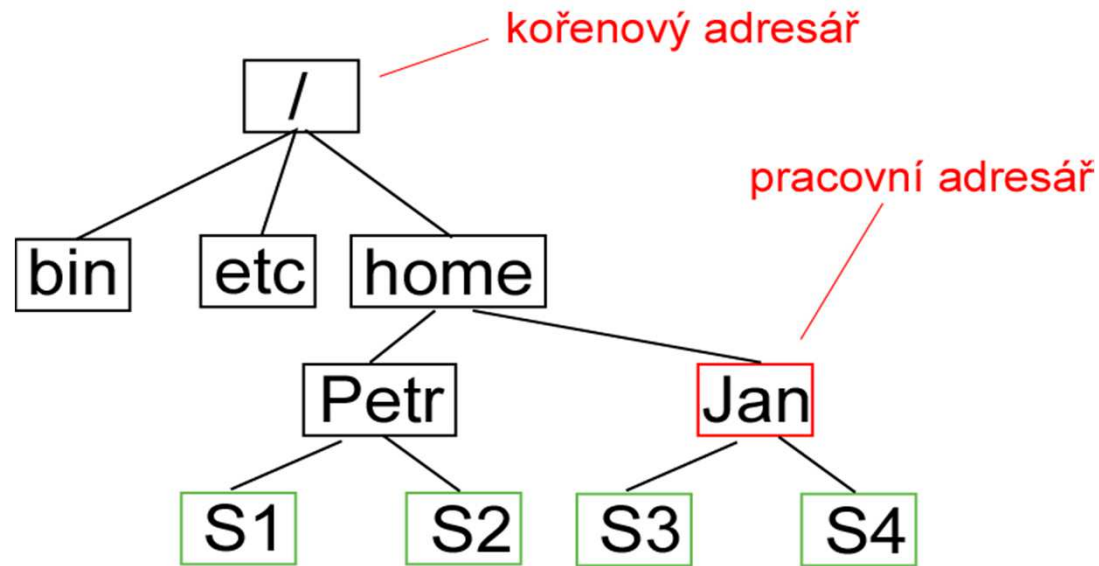
- **Relativní cesta**

- začíná vždy v pracovním (aktuálním) adresáři `$PWD`
- obsahuje posloupnost podadresářů mezi `$PWD` a cílovým souborem

```
PWD=/home/rocnik1/skupina15  
../skupina12/Novak
```

- **Domovský adresář (home directory)**

- definován pro každého uživatele
- po přihlášení je pracovním adresářem
- jeho hodnota je uložena v proměnné `HOME`
- nastaví se jako pracovní příkazem `cd` bez parametrů
- uživatel je většinou jeho vlastníkem a má právo zápisu



`/home/Petr/S1`

absolutní cesta k S1

`../../Petr/S1`

relativní cesta k S1

`../Petr/S1`

relativní cesta k S1

`/home/Jan/S4`

absolutní cesta k S4

`./S4`

relativní cesta k S4

`S4`

relativní cesta k S4

`../../bin`

relativní cesta do /bin



# Implementace FS

Rozložení dat na fyzickém disku:

Disk label + Zavaděč OS	Super blok SB	Informace o volných dat. strukturách ( i-uzlech, blocích,...)	Tabulka i-uzlů	Datové bloky (soubory a adresáře)
-------------------------------	------------------	---	-------------------	--------------------------------------

- **Disk label**
  - definuje rozdělení disku na menší oblasti (každá oblast může obsahovat jeden fyzický systém souborů)
- **Zavaděč OS**
  - nahraje jádro OS do paměti a předá mu řízení
- **Super blok**
  - obsahuje klíčové informace o celém systému souborů
- **Informace o volných datových strukturách**
- **Tabulka i-uzlů**
  - obsahuje atributy souborů a adresy dat. bloků, kde je uložen obsah souborů



# Implementace FS

Tabulka i-uzlů

	Atributy souboru	Adresy datových bloků
0	Rezervováno	
1	Rezervováno	
2	drwxr-xr-x, root, root, 2, 512, Sep 5, ...	100,
	...	...
15		
16		
17		
	...	...

Datový blok 100

Jméno souboru	Číslo i-uzlu
.	2
..	2



# Vytvoření adresáře

Tabulka i-uzlů

	Atributy souboru	Adresy datových bloků
0	Rezervováno	
1	Rezervováno	
2	drwxr-xr-x, root, root, 3, 512, Sep 5, ...	100,
	...	...
15	drwxr-xr-x, root, root, 2, 512, Sep 7,...	203,
16		
17		
	...	...

Příkazy:

\$ **mkdir /DIR**

Datový blok 100

Jméno souboru	Číslo i-uzlu
.	2
..	2
<b>DIR</b>	<b>15</b>

Datový blok 203

Jméno souboru	Číslo i-uzlu
.	15
..	2



# Vytvoření obyčejného souboru

Tabulka i-uzlů

	Atributy souboru	Adresy datových bloků
0	Rezervováno	
1	Rezervováno	
2	drwxr-xr-x, root, root, 3, 512, Sep 5, ...	100,
	...	...
15	drwxr-xr-x, root, root, 2, 512, Sep 7,...	203,
16	-rw-r--r--, root, root, 1, 512, Sep 7,...	204,
17		
	...	...

Příkazy:

\$ **mkdir /DIR**

\$ **echo "Dobre rano" > /DIR/s.txt**

Datový blok 100

Jméno souboru	Číslo i-uzlu
.	2
..	2
<b>DIR</b>	<b>15</b>

Datový blok 203

Jméno souboru	Číslo i-uzlu
.	15
..	2
<b>s.txt</b>	<b>16</b>

Datový blok 204

Dobre rano
------------



# Vytvoření pevného linku

Tabulka i-uzlů

	Atributy souboru	Adresy datových bloků
0	Rezervováno	
1	Rezervováno	
2	drwxr-xr-x, root, root, 3, 512, Sep 5, ...	100,
	...	...
15	drwxr-xr-x, root, root, 2, 512, Sep 7,...	203,
16	-rw-r--r--, root, root, 2, 512, Sep 7,...	204,
17		
	...	...

Příkazy:

\$ **mkdir** /DIR

\$ **echo** "Dobre rano" > /DIR/s.txt

\$ **ln** /DIR/s.txt /tl.txt

Datový blok 100

Jméno souboru	Číslo i-uzlu
.	2
..	2
<b>DIR</b>	<b>15</b>
tl.txt	16

Datový blok 203

Jméno souboru	Číslo i-uzlu
.	15
..	2
s.txt	16

Datový blok 204

Dobre rano
------------





## Vytvoření

`ln původní_jméno_souboru nové_jméno_souboru`

- Atributy a data jednoho souboru jsou **přístupná přes několik různých jmen**.
- Lze vytvářet pouze v **rámci jednoho fyzického systému souborů**.
- **Nelze vytvářet na**
  - adresáře
  - neexistující soubory
- Jména jsou rovnocenná (nelze zjistit, které bylo první).
- **Rušení**
  - zrušení souboru s více linky ruší pouze link a dekrementuje čítač
  - data se uvolní až při zrušení posledního linku



# Vytvoření symbolického linku

Tabulka i-uzlů

	Atributy souboru	Adresy datových bloků
0	Rezervováno	
1	Rezervováno	
2	drwxr-xr-x, root, root, 3, 512, Sep 5, ...	100,
	...	...
15	drwxr-xr-x, root, root, 2, 512, Sep 7,...	203,
16	-rw-r--r--, root, root, 2, 512, Sep 7,...	204,
17	lrwxrwx, root, root, 1, 512, Sep 7,...	205,
	...	...

Příkazy:

```
$ mkdir /DIR
```

```
$ echo "Dobre rano" > /DIR/s.txt
```

```
$ ln /DIR/s.txt /tl.txt
```

```
$ ln -s /DIR/s.txt /DIR/sl.txt
```

Datový blok 100

Jméno souboru	Číslo i-uzlu
.	2
..	2
DIR	15
tl.txt	16

Datový blok 203

Jméno souboru	Číslo i-uzlu
.	15
..	2
s.txt	16
sl.txt	17

Datový blok 204

Dobre rano
------------

Datový blok 205

/DIR/s.txt
------------



- **Vytvoření**

```
ln -s původní_jméno_souboru nové_jméno_souboru
```

- Soubor typu symbolický link obsahuje jméno jiného souboru v datovém bloku nebo přímo v i-uzlu.
- Lze vytvářet
  - i mezi **různými fyzickými systémy souborů**
  - na adresář
  - na neexistující soubory (chyba se zjistí až při použití linku)
- Při použití sym. linku se některé operace provádějí přímo s linkem (**rm**), některé se souborem, na který link ukazuje (**vi**).



# Základní příkazy: adresáře

<code>pwd</code>	vypíše jméno pracovního adresáře
<code>cd adresář</code>	změní pracovní adresář
<code>ls [-ladL] adresář</code>	vypíše obsah adresáře
<code>mkdir [-p] adresář</code>	vytvoří adresář/adresáře
<code>rmdir adresář</code>	smaže prázdný adresář
<code>rm -r adresář</code>	smaže adresář (nemusí být prázdný)

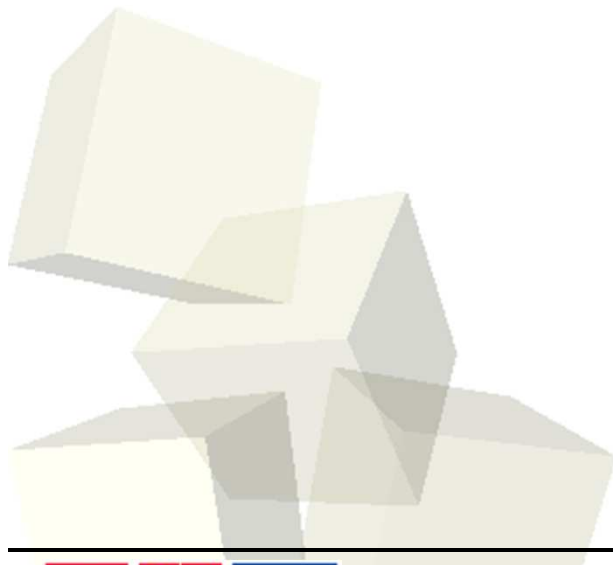


# Základní příkazy: adresáře

<code>cp -r adr1 adr2</code>	adr2 neexistuje: vytvoří kopii adr1 pojmenovanou adr2 adr2 existuje: v adr2 vytvoří kopii adr1 (adr2/adr1)
<code>mv adr1 adr2</code>	adr2 neexistuje: přejmenuje adr1 na adr2 adr2 existuje: přesune adr1 do adr2 (adr2/adr1)

- Pozn.: pozor na možnou rekurzi

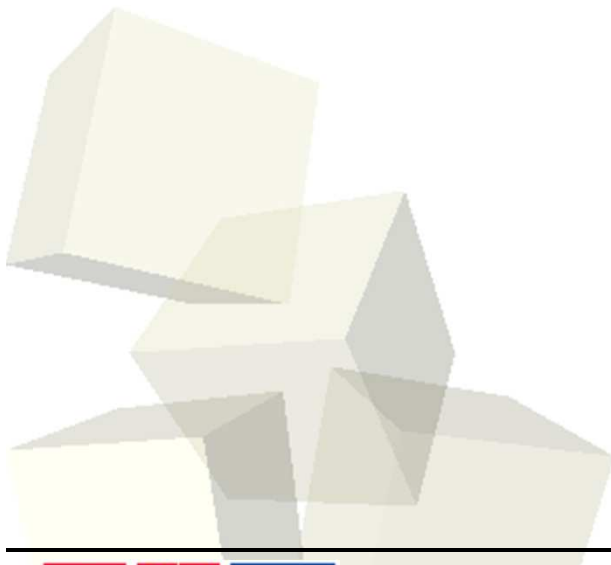
`cp -r adr1 adr1`





# Základní příkazy: soubory

<code>cp s1 s2</code>	s2 nexistuje: zkopíruje soubor s1 na soubor s2 s2 existuje: přepíše s2 souborem s1
<code>cp s1 s2 adr</code>	soubory s1 s2 zkopíruje do adresáře adr
<code>mv s1 s2</code>	přesune/přejmenuje s1 na s2
<code>rm soubor</code>	smaže soubor





# Základní příkazy: soubory

<code>file soubor</code>	vypíše informaci o typu souboru
<code>cat soubor</code>	zobrazí obsah text. souboru
<code>more soubor</code>	zobrazí obsah text. souboru po stránkách
<code>less soubor</code>	zobrazí obsah text. souboru po stránkách
<code>od -c soubor   more</code>	zobrazí obsah binárního souboru
<code>strings soubor</code>	zobrazí tisknutelné znaky z binárního souboru