

Cvičení 4.

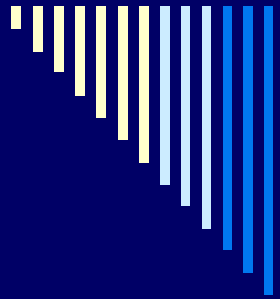
ZOS 2005, L. Pešička





Pravdivostní hodnota příkazu

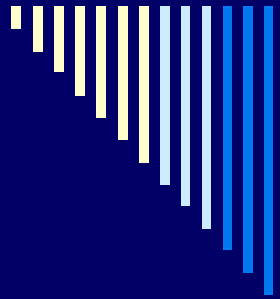
- proces po skončení předává **návratovou hodnotu** (exit status)
 - konvence
 - **0** – **úspěšné** ukončení příkazu
 - jiná hodnota – **neúspěch**
 - pokud proces ukončen signálem
 - **128+n** .. (n je číslo signálu)
-



Potomek procesu

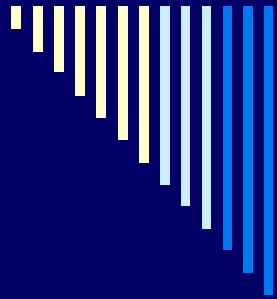
- potomek procesu zdědí **kopii prostředí** svého rodiče (proměnné, akt. adresář)
- potomek může prostředí měnit, ale změny se **nedotknou** rodiče – mění **kopii** původních dat

- pwd
- **skript** (mění pracovní adresář)
- pwd - aktuální adresář zůstal **nezměněn**



Příkazy

- **vestavěné** příkazy shellu (cd, set)
- **externí** příkazy (ls, cp, mv)
 - spustitelné soubory
 - shell je spuštěn jako svého potomka
- pokud není přeložený program
 - očekává textový soubor obsahující příkazy shellu
 - shell script, příkazový soubor
 - shell spustí svojí kopii a ta provede příkazy



Příkazový soubor

- shell zkoumá první řádek
- pokud obsahuje **#! řetězec**
 - **řetězec** – jméno programu, kterým bude soubor interpretován
 - systém tento program spustí a předá mu jméno souboru jako argument
- *#! /bin/bash*
- *echo Jsem skript pusteny bashem*



Příklad – zdravící skript

```
#!/bin/bash
```

```
echo -n "Zadej svoje jmeno: "
```

```
read PREZDIVKA
```

```
echo Nazdarek, ty jsi $PREZDIVKA
```

```
echo Prihlaseny jako $USER
```

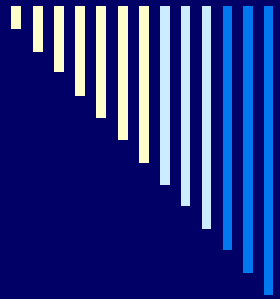
- ❑ KVIZ – příkaz set ; vidíte proměnnou PREZDIVKA?
- ❑ KVIZ – spustim tento skript: prikklad2 Lada Je rozdíl?



Podmíněné příkazy - if

```
if seznam-prikazu
then seznam-prikazu
[ elif seznam-prikazu
  then seznam-prikazu ] ...
[ else seznam-prikazu ]
fi
```

- vykoná se seznam příkazů za if, pokud návratová hodnota 0, provedou se příkazy za then



Příklad - if

```
#!/bin/bash
```

```
if test -d pivo
```

```
then
```

```
    echo "Existuje adresar pivo :)"
```

```
elif test -d vino
```

```
then
```

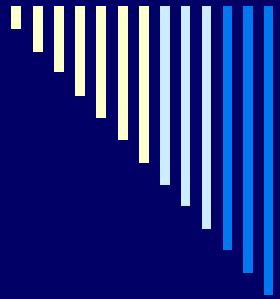
```
    echo "Neni pivo, tak alespon vino"
```

```
else
```

```
    echo "Hmm, ani pivo, ani vino"
```

```
fi
```

❑ `mkdir pivo ; prikklad3 ; mkdir vino ; prikklad3 ; rmdir pivo ; prikklad3 ..`



Podmíněný příkaz - case

case slovo **in**

vzor) seznam-prikazu ;;

vzor) seznam-prikazu ;;

...

esac

- srovnává slovo se vzorem
- pokud souhlasí, vykoná seznam příkazů a skončí



Příklad – case

```
#!/bin/sh
```

```
case $1 in
```

```
-h | -help) echo "Tady je napoveda, ze?"  
            echo "Pust me treba bez parametru."
```

```
;;
```

```
-a)        echo "Mam rad Anetu" ;;
```

```
-l)        echo "Mam rad Lenku" ;;
```

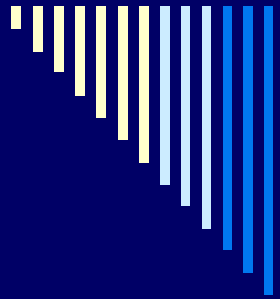
```
-k)        echo "Mam rad Kacenku" ;;
```

```
*)        echo "Chces vedet koho mam rad?"
```

```
            echo "parametry -a -l -k"
```

```
;;
```

```
esac
```



Příkaz cyklu - for

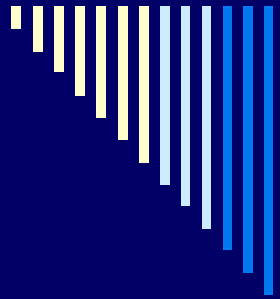
for proměnná **in** seznam-slov

do

seznam-příkazů

done

- ❑ nejdříve expandován seznam slov
- ❑ oddělena mezerou



Příklad - for

```
#!/bin/sh
```

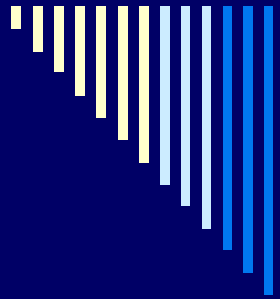
```
for F in *.kuk
```

```
do
```

```
  mv $F ${F}kuk
```

```
done
```

- najde všechny soubory *.kuk a
- přejmenuje je na *.kukkuk



Cyklus while, until

while seznam-prikazu

do seznam-prikazu

done

- ❑ dokud splněna podmínka (kod 0), cykluje
- ❑ analogický příkaz **until** false ..
- ❑ **break** – ukončení vnitřní smyčky
- ❑ **continue** – vykonání smyčky



Operátory && a ||

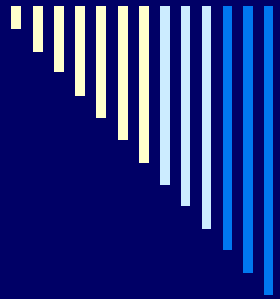
- binární operátory:

- prikaz1 && prikaz2

- vyvolá prikaz1, pokud je návratová hodnota 0 (OK), vyvolá i prikaz2

- prikaz1 || prikaz2

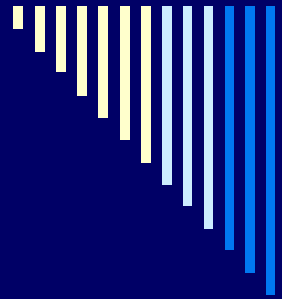
- vyvolá prikaz1, pokud je návrat. hodnota nenulová (neúspěch), vyvolá i prikaz2



Příklady

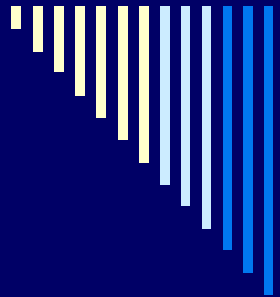
- `test -d rybnik || mkdir rybnik`
 - neexistuje-li adresář, vytvoříme
- `test -d rybnik && mv ryba rybnik/ryba`
 - je-li rybník, přesuneme do něj soubor

- vyzkoušejte z příkazové řádky
- `touch ryba ; mkdir rybnik ; atd...`



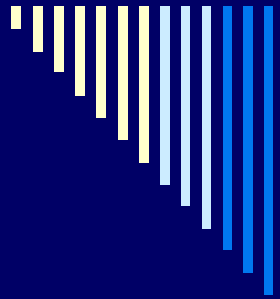
Vyhodnocení podmínky - test

```
if [ podmínka ]  
then  
    seznam-prikazu  
fi
```

Podmínky

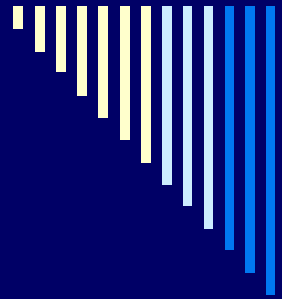
- r soubor soubor přístupný pro čtení uživateli
- w soubor soubor přístupný pro zápis
- x soubor uživatel může soubor spustit
- f soubor obyčejný soubor
- d soubor adresář
- c soubor znakový speciální
- b soubor blokový speciální
- p soubor pojmenovanou rourou
- u soubor nastavený set-user-ID bit
- g soubor nastavený set-group-ID bit
- k soubor nastavený sticky bit



Podmínky2

- -z retezec pravdivé, když délka řetězce nulová
- -n retezec pravdivé, když délka řetězce nenulová
- $r1 = r2$ pravdivé, když identické
- $r1 \neq r2$ pravdivé, když různé
- retezec pravdivé, když neprázdný

- $n1$ relační op $n2$
- -eq, -ne, -lt, -le, -gt, -ge
- ! vyraz negace
- vyraz1 -a vyraz2 oba pravdivé AND
- vyraz1 -o vyraz2 alespon jeden OR



Odkazy

- <http://rute.2038bug.com/node10.html.gz>
-