

Skripty - pokračování

Aritmetická expanze:

Používá se pro počítání (aritmetické operace) s proměnnými. Provádí operace sčítání +, odečítání -, násobení *, dělení /, zbytek po dělení %.

Formát:

`$((vyraz))`

Př.: příklady výrazů uvnitř `$((vyraz))`

`1+2`

`4/2`

`$A+1`

`$A+$B`

Př.: napište skript `dve_cisla`, který načte 2 čísla a provede jejich součet, rozdíl, součin a podíl.

```
#!/bin/bash
#Tento skript nacita 2 cisla a provadi s nimi
#aritmeticke operace pomoci aritmeticke expanze

echo "Zadej cislo A:"
read A
echo "Zadej cislo B:"
read B

echo "Soucet cisel A a B je $(( $A+$B ))"
echo "Rozdil cisel A a B je $(( $A-$B ))"
echo "Soucin cisel A a B je $(( $A*$B ))"
echo "Podil cisel A a B je $(( $A/$B ))"
```

Př.: modifikujte skript `dve_cisla` tak, aby vstupní čísla načítal pomocí pozičních parametrů...

Použití skriptu:

`dve_cisla_podruhe 4 10`

Příkaz "větvení" – IF

Formát:

```
if podminka
    then prikazy1
    else prikazy2
fi
```

Pokud je podmínka splněna, provedou se prikazy1, jinak se provedou prikazy2.

Př.: modifikujte skript telefon, který vyhledává v telefonním seznamu (v souboru seznam.txt) tak, aby v případě, že jméno v seznamu nenajde, vypsal hlášení.

```
#!/bin/ksh
#Tento skript hleda zadane jmeno v souboru seznam.txt

#Zadani jmena:
echo Zadej hledane jmeno
read jmeno
#Kontrolni vypis promenne, zda se ulozila spravne
echo Myslis opravdu toto jmeno?
echo $jmeno

#samotne vyhledavani v souboru seznam.txt
#s podmínkou, která vypisuje hlaseni
if grep $jmeno seznam.txt >/dev/null
    then echo "Telefon je: `grep $jmeno seznam.txt`"
    else echo "Toto jmeno $jmeno neznam!"
fi
```

Jak tvořit podmínku – příkaz test

Formát:

[podmínka]

Př.: příklady podmínek uvnitř []. Více v "man test "

-f soubor	soubor existuje
-r soubor	soubor existuje a lze jej číst
řetězec1 = řetězec2	řetězec1 je shodný s řetězcem2
řetězec1 != řetězec2	řetězce se neshodují
-z řetězec	řetězec má nulovou délku
číslo1 -eq číslo2	číslo1 = číslo2 (equal)
číslo1 -ne číslo2	číslo1 se nerovná číslo2 (not equal)
číslo1 -lt číslo2	číslo1 < číslo2 (lower than)

analog. -gt (greater than), -le (lower or equal) -ge (greater or equal)...

Př.: skript, který otestuje, zda jste v domovském adresáři, a podle výsledku vypíše buď "Jste doma!" anebo "Jste v adresáři..."

```
#!/bin/bash
#tento skript testuje, v jakem adresari
#se nachazim

if [ `pwd` = $HOME ]
    then echo "Jste doma"
    else echo "Jste v adresari "
        pwd
fi
```

Př.: skript, který vypíše soubor uvedený jako první (poziční) parametr programu. Pokud soubor nelze vypsát, vypíše se hlášení.

```
#!/bin/bash
#tento skript vypisuje soubor anebo hlasi,
#ze to nejde

if [ -r $1 ]
    then cat $1
    else echo "Soubor $1 nelze vypsát"
fi
```

Příkaz "cyklu" – FOR

Formát:

```
for promenna in seznam
do
    prikazy
done
```

Pro všechny položky v seznamu vykonává definovanou činnost.

Př.: skript vypisující obsah 3 proměnných.

```
#!/bin/bash
#
for P in "a b c"
do
    echo $P
done
```

Seznam lze vytvořit i vykonáním nějakého příkazu, např. `ls`, `cat`.

Př.: skript kolečko, který vypíše postupně řádky souboru `kolo.seznam`

```
#!/bin/bash
#
for i in `cat kolo.seznam`
do
    echo Tohle je $i
done
```

Příkaz "cyklu s podmínkou" – WHILE

Formát:

```
while podmínka
do
    příkazy
done
```

"pokud platí, že..." (pokud návratový kód = 0).

Podmínka se tvoří stejně jako pro příkaz if. Dokud podmínka platí, provádí se příkazy. V okamžiku, kdy přestane platit, přestanou se provádět.

Př.: skript kolecko, který vypíše postupně řádky souboru kolo.seznam

```
#!/bin/bash
#skript, který napocita do 10

start=1
end=1
echo "Zacnu na cisle $start "
echo "a skoncim na cisle $end "
while [ $start -le $end ]
do
    echo $start
    start ==$((start+1))
done
```

Analog. **Příkaz "cyklu s podmínkou" – UNTIL**

vykonává příkazy, pokud podmínka **NENÍ** splněna, návratový kód je = 1 (false).
Jakmile podmínka **JE** splněna, cyklus končí.

"Vícecestná" podmínka – CASE

```
case text in
  text1) prikazy;;
  text2) prikazy;;
  ...
  textn) prikazy;;
esac
```

Př.: skript budík, který v závislosti na denní či noční době vypisuje hlášky.

```
#!/bin/bash
h=`date +"%l" `
#echo $h

case $h in
  0*) echo "Sladke sny...";;
  ?6*) echo "Je cas vstavat!";;
  ?7*) echo "Ale uz vazne vstavej!";;
  ?9) echo "Odchod do UNIXu!";;
  10) echo "Jeste stale programujeme...";;
  12) echo "Cas na obed!";;
  18) echo "Jdeme domu...";;
  19) echo "Vecernicek";;
  2*) echo "Jdi spat!";;
  *) echo "UNIX Forever!";;
esac
```