

# A7B38UOS Úvod do operačních systémů

## 10. Cvičení Skripty

# Obsah cvičení

**Zpracování argumentů**

**Zpracování uživatelského vstupu**

**Funkce**

# Spuštění skriptu

- **v aktuální instanci shellu**
  - `skript` (min. práva skriptu r--)
- **v nové instanci shellu**
  - `/bin/sh skript` (min. práva skriptu r--)
  - `./skript` (min. práva skriptu r-x)

# Ladění skriptu

- **v-mód** = původní řádka ze skriptu se nejdříve vytiskne na standardní výstup a pak se spustí
- **x-mod** = provedou se náhrady spec. znaků na řádce, vytiskne se na standardní výstup a pak se spustí
- **Ladění celého skriptu**
  - spustíme shell s parametrem **-v**
  - spustíme shell s parametrem **-x**
- **Ladění části skriptu**
  - povolení ladicích výpisů pomocí **set -v** a **set -x**
  - zakázání ladicích výpisů pomocí **set +v** a **set +x**

# Argumenty skriptu

- Při spouštění skriptu se jméno skriptu a argumenty na příkazové řádce uloží do následujících proměnných:

Jméno proměnné	Význam proměnné
<code>\$#</code>	Počet parametrů skriptu
<code>\$0</code>	Jméno skriptu
<code>\$1, ..., \$9, \${10}, ...</code>	Argumenty skriptu (sh pouze \$1,...,\$9)
<code>\$*</code>	<code>\$1 \$2 \$3 ...</code>
<code>\$@</code>	<code>\$1 \$2 \$3 ...</code>
<code>"\$*"</code>	<code>"\$1 \$2 \$3 ..."</code>
<code>"\$@"</code>	<code>"\$1" "\$2" "\$3" ...</code>

# Argumenty skriptu

- **Příkaz: set -- seznam argumentu**
  - rozdělí seznam argumentů podle \$IFS a přiřadí je do \$1, \$2, \$3...
- **Příkaz: shift [n]**
  - posune hodnoty argumentů vlevo:  $\$i = \${i+n}$
  - odebere n argumentu z \$\* a \$@
  - dekrementuje:  $\$# = \$# - n$

# Argumenty skriptu

## Skript param1.sh

```
#!/bin/sh
echo "Jmeno skriptu: $0"
echo "Pocet argumentu: $# "
echo "Hodnota \"\$*\": @$"

echo
echo "Hodnota 1. argumentu: $1"
echo "Hodnota 2. argumentu: $2"
```

# Argumenty skriptu

## Skript param2.sh

```
#!/bin/sh

echo "Pocet argumentu: $#"
```

```
while [ $# -gt 0 ]
do
    echo "Hodnota \$#: $#"
```

```
    echo "Hodnota \$*: $*"
```

```
    echo "Hodnota \$@: @$"
```

```
    echo "Hodnota \$1: $1"
```

```
    echo
```

```
    shift
done
```



# Příkaz read

**read P1 P2 P3**

- Přečte jednu řádku ze vstupu.
- Podle proměnné **\$IFS** rozdělí načtenou řádku na jednotlivé hodnoty.
- Uloží první hodnotu do proměnné **P1**, druhou položku do proměnné **P2** a ostatní hodnoty do proměnné **P3** .

# Příkaz read - příklad

## Skript read.sh

```
#!/bin/sh
echo "Informace o uzivatelich v /etc/passwd"
IFS=":"

while read JMENO X UID GID POPIS DIR LOGSHELL
do
    echo "Ucet $JMENO ma:"
    echo "UID=$UID"
    echo "GID=$GID"
    echo "HOME=$DIR"
    echo "SHELL=${LOGSHELL:-neni definovan}"
done < /etc/passwd
```

# Funkce

## Syntaxe

```
[function] jméno_funkce  
{  
    seznam_příkazů  
}
```

## Popis

- Funkce se volá jménem a volitelnými parametry
- Počet a hodnoty parametrů jsou uvnitř funkce dostupné pomocí \$# a pozičních parametrů \$1, \$2, ...
- Funkce může volat jinou funkci případně sebe rekurzivně.
- Návrat z funkce proběhne po provedení všech příkazů s návratovým kódem posledního provedeného příkazu
- nebo předčasně příkazem **return návratový\_kód**.

# Funkce

## Skript max.sh

```
maximum()  
{  
  if [ $1 -gt $2 ] ; then  
    echo "$1"  
  else  
    echo "$2"  
  fi  
}  
  
if [ $# -ne 2 ] ; then  
  echo "Pouziti: $0 num1 num2"  
  exit 1  
fi  
echo "Maximun je `maximum $1 $2` "
```