

## Ćwiczenie 4. Skrypty powłoki. Parametry skryptów. Instrukcja if.

### 1. Zaloguj się w systemie Unix.

### 2. Uruchamianie prostych skryptów.

- Zapisz do pliku o nazwie **p1** ciąg poleceń:

```
echo Dzisiaj jest date  
echo Dzisiaj jest `date`  
echo Liczba uzytkownikow: `who|wc -l`  
echo Jestem: `who am i`
```

#### Zadanie 1

Napisz skrypt (zapisz go w pliku **z1**), który wyświetla nazwę katalogu bieżącego oraz liczbę zapisanych w nim plików zwykłych i liczbę podkatalogów. Do wyświetlenia nazwy katalogu wykorzystaj polecenie pwd umieszczone w odwrotnych apostrofach ( jak 2 wiersz skryptu **p1**). Do wyznaczania liczby plików wykorzystaj polecenia: **ls**, **grep** i **wc**.

#### Zadanie 2

Opracuj skrypt **z2** prezentujący zawartość katalogu bieżącego wg wzoru:

```
Katalog o nazwie: nazwa_katalogu zawiera:  
Katalogi o nazwach:  
katalog1  
katalog2 itd.  
Pliki:  
Wielkosc      Nazwa  
rozmiar1     plik1  
rozmiar2     plik2 itd.
```

Wykorzystaj polecenia **ls**, **grep** i **cut**.

### 3. Parametry skryptów

- Zapisz do pliku o nazwie **p2** ciąg poleceń:

```
echo Pierwszy parametr skryptu: $1  
echo Drugi parametr skryptu: $2  
echo Trzeci parametr skryptu: $3  
echo Wszystkie parametry skryptu: $*  
echo Jeszcze raz wszystkie parametry skryptu: @$  
echo Liczba parametrów: $#
```

Przetestuj skrypt **p2** uruchamiając go z jednym, dwoma, trzema i czterema dowolnymi parametrami.

- Zapisz do pliku o nazwie **p3** ciąg poleceń:

```
echo Ten skrypt wyświetla plik o nazwie określonej pierwszym parametrem  
echo a następnie tworzy katalog o nazwie określonej drugim parametrem  
echo i kopiuje do niego plik  
echo Tresc pliku o nazwie: $1  
cat $1
```

```
mkdir $2
cp $1 $2
cd $2
echo w katalogu `pwd` znajduje sie plik: `ls`
```

Przetestuj skrypt **p3** uruchamiając go z poleceniem: **p3 p2 kat**. Sprawdź czy skrypt utworzył katalog **kat** i skopiował do niego plik **p2**.

### Zadanie 3

Napisz skrypt **z3**, który zakłada katalog o nazwie określonej pierwszym parametrem i umieszcza w nim dwa pliki o nazwach określonych kolejnymi parametrami. Zawartość pierwszego pliku mają stanowić nazwy użytkowników (pierwsza kolumna pliku `/etc/passwd`), zawartość drugiego nazwy grup (pierwsza kolumna pliku `/etc/group`).

### 4. Polecenie przesuwania parametrów: shift

- Zapisz do pliku o nazwie **p4** ciąg poleceń:

```
echo Parametr \${1}: ${1}
echo Parametr \${2}: ${2}
echo Wszystkie parametry skryptu: $*
shift
echo wykonano polecenie shift
echo 'Parametr $1:' ${1}
echo 'Parametr $2:' ${2}
echo Wszystkie parametry skryptu: $*
```

Przetestuj skrypt **p4** uruchamiając go z trzema dowolnymi parametrami.

### Zadanie 4

Napisz skrypt **z4**, który zakłada katalog o nazwie określonej pierwszym parametrem i umieszcza w nim puste pliki o nazwach określonych pozostałymi parametrami ( w dowolnej liczbie). Do wygenerowania plików zastosuj polecenie `touch`.

### 5. Instrukcja if

- Zapisz do pliku **p5** ciąg poleceń:

```
if [ "$1" == "" ] # Uwaga! prawidłowo rozmieść spacje w nawiasach [ ]
then echo Brak parametru!
else echo Podano parametr: "$1"
fi
```

Przetestuj skrypt **p5** uruchamiając go z parametrem i bez parametru.

- Zapisz do pliku **p6** ciąg poleceń:

```
if [ -f $1 ]
then echo "$1 jest plikiem"
elif [ -d $1 ]
then echo $1 jest katalogiem
else echo $1 nie jest ani plikiem ani katalogiem
fi
```

Przetestuj skrypt uruchamiając go bez parametru, z parametrem - nazwą pliku oraz z parametrem – dowolnym wyrazem.

### Zadanie 5

Opracuj skrypt o nazwie **z5** wyświetlający informację o obecności lub braku parametru, a następnie, w przypadku gdy parametr występuje informację czy parametr jest nazwą pliku czy katalogu.

### Zadanie 6

Opracuj skrypt o nazwie **z6** wyświetlający komunikat zależny od liczby parametrów skryptu:

0 parametrów:	Nie podano parametrow
1 parametr:	Jest 1 parametr
2, 3 lub 4 parametry:	Sa \$# parametry
5 lub więcej parametrów:	Jest \$# parametrow

### Zadanie 7

Zmodyfikuj skrypt **z3** (rozwiązanie zadania 3) w ten sposób by nie powodował błędu wykonania polecenia **mkdir** gdy zakładany katalog już istnieje. Skrypt powinien także kończyć działanie (z odpowiednim komunikatem) jeśli w katalogu istnieją już pliki o nazwach podanych parametrem drugim i trzecim.

- Zapisz do pliku **p7** ciąg poleceń:

```
if grep $1 $2 >/dev/null
then
    echo Kod zakonczenia ostatniego polecenia: $?
    echo oznacza to ze polecenie: " grep $1 $2 " zostalo zakonczone sukcesem
    echo tj. znaleziono slowo $1 w pliku $2
else
    echo Kod zakonczenia ostatniego polecenia: $?
    echo oznacza to ze polecenie: " grep $1 $2 " zostalo zakonczone niepowodzeniem
    echo tj. w pliku $2 nie ma slowa $1
fi
```

Przetestuj działanie skryptu wywołując go: **p7 stal p7** oraz **p7 nikiel p7**

### Zadanie 8

Napisz skrypt **z8**, który wyświetla (przy wykorzystaniu polecenia **cat**) treść pliku określonego parametrem. W przypadku gdy plik nie istnieje lub jest zabezpieczony przed czytaniem skrypt powinien wyświetlać odpowiedni komunikat.