

Systemy Operacyjne – laboratorium Zachodniopomorska Szkoła Biznesu			
Grupa/Specialność	Imię i nazwisko	Data wykonania (rrrr.mm.dd)	Tryb studiów
Game Design	Sylwester Dawidowicz	3.06.2020	Z
Nr laboratorium	Temat		
5	Bash		

Na tych laboratoriach były do wykonania 3 zadania z wykorzystaniem języka skryptowego PowerShell. W następnej części sprawozdania opiszę poszczególne z nich.

**ZADANIE 1:** Napisać skrypt, który pobierze od użytkownika nazwę folderu i liczbę. Następnie utworzy on folder o takiej nazwie, a w nim tyle plików, ile użytkownik podał jako parametr. Należy skorzystać z metody `read`, polecam użyć flagi `-p` do wyświetlenia komunikatu zachęty

Kod programu wygląda następująco:

```
#!/bin/bash

#Pobieramy nazwę folderu od użytkownika
read -p "Podaj nazwę folderu, który zostanie stworzony: " folder
files=-1;

#Sprawdzamy, czy liczba podana przez użytkownika jest większa od zera
while [ $files -le 0 ] ; do
    read -p "Podaj liczbę większą od zero: " files
done

#Tworzymy nowy folder o nazwie podanej przez użytkownika
mkdir $folder

#Tworzymy ilość plików odpowiadających liczbie podanej przez użytkownika
for (( i=1; i <= files; i++)) ; do
    touch ./$folder/$i.txt
done
```

Konieczne było wykorzystanie tutaj polecenia `read` z flagą `-p` proszącego użytkownika komunikatem zaczęty o podanie zmiennych. Następnie poleceniem `mkdir` tworzymy nasz nowy katalog i dalej w pętli za pomocą `touch` tworzymy kolejne pliki w tym folderze. Wynik działania programu:

```
sylwester@sylwester-Legion-Y540-15IRH: ~/Dokumenty
sylwester@sylwester-Legion-Y540-15IRH:~/Dokumenty$ ./lab3_zadanie1
Podaj nazwę folderu, który zostanie stworzony: nowy
Podaj liczbę większą od zero: -4
Podaj liczbę większą od zero: 7
sylwester@sylwester-Legion-Y540-15IRH:~/Dokumenty$
```

**ZADANIE 2:** Napisać skrypt, który przyjmie w linii komend terminala nazwę pliku oraz znak. Skrypt ma za zadanie wyświetlić informację, ile razy dany znak występuje w podanym pliku.

Kod programu wygląda następująco:

```
#!/bin/bash

#Wprowadzamy możliwość czytania parametrów z linii komend i definiujemy odpowiednie parametry
while getopts c:f: option
do
    case "${option}"
    in
        f) file_name=${OPTARG};;
        c) user_char=${OPTARG};;
    esac
done

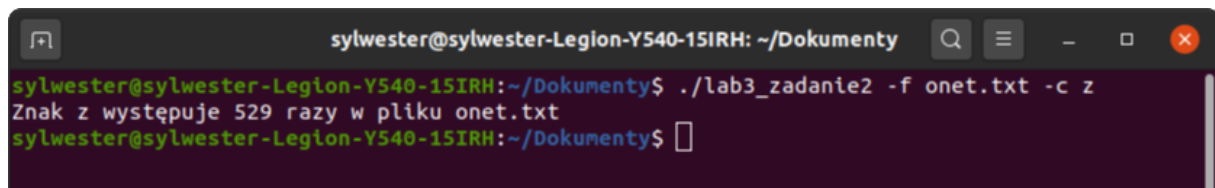
#Sprawdzanie, czy plik istnieje
if [ -f "$file_name" ] ; then

#Funkcja zliczająca ilość wystąpień danego znaku
count="$(grep -o "$user_char" $file_name | wc -l)"

#Wypisanie wystąpień podanego znaku
echo "Znak $user_char występuje $count razy w pliku $file_name"

#Wyświetlenie komunikatu, jeżeli plik nie istnieje
elif [ ! -f "$file_name" ] ; then
    echo "Plik o podanej nazwie nie istnieje!"
fi
```

Z linii komend uruchamiamy program z zdefiniowanymi flagami f (dla ścieżki do pliku) oraz c (dla szukanego znaku). Następnie sprawdzamy, czy plik istnieje, zliczamy ilość wystąpień znaku oraz wypisujemy wynik:

A terminal window with a dark background. The title bar shows the user 'sylwester' on a machine named 'sylwester-Legion-Y540-15IRH' in the directory '~/Dokumenty'. The terminal shows a command being executed: './lab3\_zadanie2 -f onet.txt -c z'. The output of the command is 'Znak z występuje 529 razy w pliku onet.txt'. The prompt returns to the shell.

```
sylwester@sylwester-Legion-Y540-15IRH: ~/Dokumenty
sylwester@sylwester-Legion-Y540-15IRH:~/Dokumenty$ ./lab3_zadanie2 -f onet.txt -c z
Znak z występuje 529 razy w pliku onet.txt
sylwester@sylwester-Legion-Y540-15IRH:~/Dokumenty$
```

Opisane zadania jak i język Bash nie sprawiły mi żadnych problemów. Polecenia jak i składnia są proste (moim subiektywnym zdaniem).