

**LISTA ZADAŃ NR 4 – SKRYPTY, POLECENIE READ,
INSTRUKCJA WARUNKOWA IF-ELIF-ELSE**

UWAGI DO ZADAŃ (DO OMÓWIENIA PRZED ROZWIĄZYWANIEM):

1) Polecenia konsoli możemy pisać w plikach tekstowych, a następnie uruchamiać je wszystkie poprzez wywołanie nazwy tego pliku. Tworzymy w ten sposób **SKRYPTY**. Utworzony skrypt uruchamiany poleceniem **bash**. Oto przykład skryptu, który tworzy 20 katalogów, a w nich 20 plików tekstowych:

```
i=1
while [ $i -le 20 ]
do
    mkdir $i
    cd $i
    touch $i.txt
    cd ..
    i=$((i+1))
done
```

Zapisz ten skrypt w pliku skrypt na Pulpicie, a następnie uruchom go poleceniem **bash** skrypt.

2) Polecenie **read** służy do wczytywania danych z klawiatury. Oto przykład prostego skryptu dodającego dwie liczby wprowadzone przez użytkownika, który możemy uruchomić w konsoli.

```
echo Podaj pierwszą liczbę
read a
echo Podaj drugą liczbę
read b
echo Suma liczby $a oraz $b wynosi $((a+b))
```

Zapisz ten skrypt w dowolnym pliku na pulpicie, a następnie uruchom go jako skrypt.

3) Jedną z najważniejszych instrukcji wykorzystywanych w terminalu jest instrukcja warunkowa **IF-ELIF-ELSE**. Jej konstrukcja jest następująca:

```
if [ warunek ]
then
    instrukcje
elif [ warunek ]
then
    instrukcje
else
    instrukcje
fi
```



dr Marcin Ziółkowski

Oto kilka przykładów:

1) Skrypt realizujący dzielenie liczb:

```
echo Podaj pierwszą liczbę
read a
echo Podaj drugą liczbę
read b
if [ $b = 0 ]
then
    echo Błędny dzielnik
else
    Iloraz liczby $a oraz $b wynosi  $[/math>
fi$ 
```

2) Skrypt kasujący niepuste pliki z zestawu plików 1.txt,...,200.txt

```
i=1
while [ $i -le 200 ]
do
    if [ -s $i.txt ]
    then
        rm $i.txt
    fi
    i=$((i+1))
done
```

Utwórz 20 plików na Pulpicie, a następnie w wybranych pięciu wpisz jakiś tekst. Sprawdź działanie powyższego skryptu.

3) Skrypt sprawdzający znak liczby wprowadzonej przez użytkownika

```
echo Podaj liczbę
read a
if [ $a -lt 0 ]
then
    echo Liczba $a$ jest ujemna
elif [ $a -gt 0 ]
then
    echo Liczba $a$ jest dodatnia
else
    echo Liczba $a$ jest zerem
fi
```

Zapisz ten skrypt w dowolnym pliku, a następnie go uruchom.



Zadanie 1. Napisz skrypt, który mnoży dwie liczby całkowite wprowadzone przez użytkownika. W poleceniach skryptu do wczytywania liczb z klawiatury użyj polecenia **read**.

Wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt1
Podaj pierwszą liczbę
5
Podaj drugą liczbę
6
Iloczyn liczb 5 oraz 6 wynosi 30
marionesta@marionesta-X501A1 ~/Pulpit $
```

Zadanie 2. Napisz skrypt, który porównuje dwie liczby całkowite wprowadzone przez użytkownika. W poleceniach skryptu do wczytywania liczb z klawiatury użyj polecenia **read**. Wykorzystaj instrukcję warunkową **if-elif-else**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
Podaj pierwszą liczbę
5
Podaj drugą liczbę
6
Większa jest druga liczba
marionesta@marionesta-X501A1 ~/Pulpit $
```

Zadanie 3. Napisz skrypt, który oblicza kwadrat liczby całkowitej wprowadzonej przez użytkownika. W poleceniach skryptu do wczytywania liczby z klawiatury użyj polecenia **read**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
Podaj liczbę
11
Kwadrat liczby 11 wynosi 121
marionesta@marionesta-X501A1 ~/Pulpit $
```



Zadanie 4. Napisz skrypt, który oblicza sumę liczb wprowadzonych przez użytkownika. Wykorzystać pętlę **while** oraz instrukcję **read**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
```

```
Ile liczb chcesz wprowadzić?
```

```
3
```

```
Podaj 1 liczbę
```

```
5
```

```
Podaj 2 liczbę
```

```
11
```

```
Podaj 3 liczbę
```

```
26
```

```
Suma wprowadzonych liczb wynosi 42
```

```
marionesta@marionesta-X501A1 ~/Pulpit $
```

Zadanie 5. Napisz skrypt, który oblicza iloczyn liczb wprowadzonych przez użytkownika. Wykorzystać pętlę **while** oraz instrukcję **read**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
```

```
Ile liczb chcesz wprowadzić?
```

```
4
```

```
Podaj 1 liczbę
```

```
5
```

```
Podaj 2 liczbę
```

```
2
```

```
Podaj 3 liczbę
```

```
3
```

```
Podaj 4 liczbę
```

```
4
```

```
Iloczyn wprowadzonych liczb wynosi 120
```

```
marionesta@marionesta-X501A1 ~/Pulpit $
```

Zadanie 6. Napisz skrypt, który tworzy kilka katalogów o nazwach zaproponowanych przez użytkownika. Wykorzystać pętlę **while** oraz instrukcję **read**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
```

```
Ile katalogów tworzysz?
```

```
2
```

```
Podaj nazwę 1 katalogu
```

```
EWA
```

```
Podaj nazwę 2 katalogu
```

```
ADAM
```

```
marionesta@marionesta-X501A1 ~/Pulpit $
```



dr Marcin Ziółkowski

Listy zadań dla studentów Akademii im. Jana Długosza w Częstochowie

Copyright by Marcin Ziółkowski & Akademia Jana Długosza 2017

Zadanie 7. Napisać skrypt **TWORZENIE**, który odpowiada za utworzenie 100 plików tekstowych na Pulpicie o nazwach 1.txt,..., 100.txt oraz na wpisaniu do plików od 1.txt do 50.txt ich nazw, następnie napisać drugi skrypt **KOPIOWANIE**, który przekopiuje zawartość pliku 1.txt do pliku 51.txt, zawartość pliku 2.txt do pliku 52.txt itd. aż do przekopiowania zawartości pliku 50.txt do pliku 100.txt. Wykorzystać instrukcję iteracyjną **while**, do kopiowania zawartości plików użyć polecenia **cat** i znaku przekierowania **>>**.

Zadanie 8. Napisać skrypt, który wyświetli serię kwadratów liczb naturalnych począwszy od liczby jeden. O długości serii decyduje użytkownik. Wykorzystać instrukcję **while** oraz polecenie **read**.

Przykładowe wykonanie skryptu powinno wyglądać następująco:

```
marionesta@marionesta-X501A1 ~/Pulpit $ bash skrypt
```

```
Ile kwadratów wyświetlamy?
```

```
5
```

```
1
```

```
4
```

```
9
```

```
16
```

```
25
```

```
marionesta@marionesta-X501A1 ~/Pulpit $
```

Zadanie 9. Napisać skrypt, który wyświetli serię sześciąt liczb naturalnych począwszy od liczby jeden. O długości serii decyduje użytkownik. Wykorzystać instrukcję **while** oraz polecenie **read**.

Zadanie 10. Napisać skrypt, który w pliku tekstowym tworzy wizytówkę użytkownika składającą się z imienia, nazwiska oraz adresu mailowego użytkownika. Wszystkie dane powinny być podawane przez użytkownika w terminalu, a następnie zapisywane w pliku tekstowym, którego nazwa powinna pokrywać się z nazwiskiem. Na przykład, jeśli wprowadzane są dane użytkownika Kamil Nowak, dane te powinny zapisywać się w pliku tekstowym Nowak.txt. Wykorzystać instrukcję **read** oraz operator przekierowania **>>**.

Zadanie 11. Napisz skrypt **KASOWANIE**, który kasuje niepuste pliki o rozszerzeniu .txt. Wykorzystać instrukcję warunkową **if-elif-else** oraz sprawdzić działanie skryptu na dziesięciu stworzonych wcześniej plikach, z których cztery są niepuste. Wskazówka: w instrukcji warunkowej **if** użyć polecenia **-s**, które oznacza "plik nie jest pusty".



dr Marcin Ziółkowski

Listy zadań dla studentów Akademii im. Jana Długosza w Częstochowie

Copyright by Marcin Ziółkowski & Akademia Jana Długosza 2017