

LISTA ZADAŃ NR 3 – WYKORZYSTANIE POLECENIA ECHO, OPERATORY ARYTMETYCZNE, PRZEKIEROWYWANIE WYJŚCIA, PĘTLA WHILE

UWAGI DO ZADAŃ (DO OMÓWIENIA PRZED ROZWIĄZYWANIEM):

1) Polecenie **echo** zwraca na wyjściu terminala swój argument, oznacza to, że jeśli wpiszemy w terminalu np. **echo Hello world**, to w terminalu wyświetli się napis **Hello world**, polecenie to jest jednym z najważniejszych poleceń w terminalu, ponieważ pozwala na edytowanie tekstów, a w połączeniu z innymi poleceniami terminala pozwala na znacznie więcej. Każdy ciąg znaków w terminalu jest interpretowany jako komenda (którą terminal próbuje wykonać) i gdyby nie sprytne polecenie **echo**, nie byłoby możliwości pisania w terminalu

2) Terminal ma wbudowane możliwości wykonywania obliczeń całkowitoliczbowych z wykorzystaniem czterech podstawowych działań arytmetycznych, wykorzystujemy tu operator **\$**

Przykłady:

a) Chcemy wykonać działanie $23 + 45$

Piszemy w terminalu **echo \$[23+45]**

b) Chcemy wykonać działanie $230 - 45$

Piszemy w terminalu **echo \$[230-45]**

c) Chcemy wykonać działanie $25 * 25$

Piszemy w terminalu **echo \$[25*25]**

d) Chcemy wykonać działanie $2000 : 40$

Piszemy w terminalu **echo \$[2000/40]**

3) Standardowym wyjściem terminala (czyli urządzeniem, na które zwracana jest wartość komendy) jest sam terminal, można jednak to zmienić przekierowując np. wyjście do pliku, służą do tego znaki przekierowania **>** oraz **>>**. Znak **>>** dopisuje na końcu pliku, nie zmieniając jego dotychczasowej zawartości, natomiast znak **>** nadpisuje plik, to znaczy zmienia dotychczasową zawartość pliku na tą wprowadzaną z wyjścia terminala. Razem z poleceniem **echo** można w ten sposób pisać do pliku **bez użycia żadnego edytora tekstowego**.



Przykłady:

echo Witajcie w naszej bajce > 1.txt - polecenie to wpisze do pliku 1.txt podany tekst, jeśli plik 1.txt nie istnieje, najpierw go stworzy

echo Słoń zagra na fujarce >> 1.txt - polecenie to wpisze do pliku 1.txt na koniec podany tekst

4) Bardzo istotnym elementem pozwalającym na pisanie programów w terminalu jest możliwość wykorzystania tzw. pętli, czyli instrukcji, które pozwalają na wielokrotne powtarzanie tych samych operacji. Najważniejszą pętlą jest pętla **WHILE**, której konstrukcja jest następująca:

```
while [ warunek kontynuacji pętli ]  
do  
    instrukcje  
done
```

Uwaga. Odstęp po nawiasie "[" oraz przed nawiasem "]" są konieczne.

Oto kilka przykładów:

a) Tworzenie serii katalogów (katalogi numerowane od 1 do 100)

```
i=1  
while [ $i -le 100 ]  
do  
    mkdir $i  
    i=$((i+1))  
done
```

b) Kasowanie katalogów o numerach od 20 do 80

```
i=20  
while [ $i -le 80 ]  
do  
    rmdir $i  
    i=$((i+1))  
done
```

c) Tworzenie plików o nazwach 1.txt do 100.txt

```
i=1  
while [ $i -le 100 ]  
do  
    touch $i.txt  
    i=$((i+1))  
done
```



Zadanie 1. Utwórz na Pulpicie katalog WIERSZ, a następnie utwórz w nim plik tekstowy wiersz.txt. Przy pomocy polecenia **echo** oraz operatora **>>** wpisz do niego dowolny tekst. W ten sam sposób dopisz dowolny tekst z użyciem operatora **>>**, a potem operatora **>**. Jaka jest różnica.

Zadanie 2. Wykorzystaj terminal jako kalkulator i oblicz wyniki następujących działań:

a) $345 + 3456$

b) $2345 - 987$

c) $345 * 543$

d) $2245 : 5$

Zadanie 3. Utwórz na Pulpicie 100 katalogów o nazwach STUDENT1,..., STUDENT100, następnie skasuj katalogi o numerach od 31 do 71. Wykorzystaj (dwa razy) instrukcję WHILE.

Zadanie 4. Utwórz na Pulpicie 100 plików o nazwach 1.txt, ..., 100.txt, następnie skasuj pliki od 22.txt do 88.txt. Wykorzystaj (dwa razy) instrukcję WHILE.

Zadanie 5. Utwórz na Pulpicie 100 katalogów o nazwach 1,...,100, następnie skasuj katalogi o numerach nieparzystych. Wykorzystaj instrukcję WHILE.

Zadanie 6. Utwórz na Pulpicie 100 katalogów o nazwach 1,...,100, następnie zmień nazwy katalogów o numerach parzystych na "A2", "A4" itd. Wykorzystaj instrukcję WHILE.

Zadanie 7. Stwórz na Pulpicie 20 katalogów o nazwach katalog1, ..., katalog20. W każdym z nich stwórz plik tekstowy odpowiadający numerowi katalogu, tzn. W katalogu1 ma być plik 1.txt, w katalogu2 ma być plik 2.txt itd. Do plików wpisz ich nazwy.



dr Marcin Ziółkowski

Listy zadań dla studentów Akademii im. Jana Długosza w Częstochowie

Copyright by Marcin Ziółkowski & Akademia Jana Długosza 2017