

Politechnika Wrocławska
Wydział Informatyki i Telekomunikacji
Systemy Operacyjne Laboratoria
Grupa 15 Tygodnie parzyste

Sprawozdanie z zajęć z dnia
07.11.2025
Zadanie nr. 3

Autor:
Daniel Gościński 280878

Prowadzący:
dr. inż. Mariusz Makuchowski

Spis treści

1. Treść zadania	3
2. Proces wykonania zadania	3
3. Wnioski	4
4. Literatura	4

1. Treść zadania

Numerowanie wszystkich plików w zadanym katalogu (parametr wywołania skryptu), do których mamy ustawione prawo wykonywania („execute”), przez dodanie dodatkowego członu rozszerzenia o postaci `.<numer kolejny>`. Numeracja powinna przebiegać według wielkości plików.

2. Proces wykonania zadania

Zadanie znajdujące się w pliku **zad1.sh** oczekiwało od nas zmiany nazwy wszystkich plików, do których mamy prawa **executable**, dodając na koniec nazwy rozszerzenie w postaci `.<numer>`. Pliki wowinny być numerowane według wielkości rozmiaru plików.

Skrypt zaczyna się od **#!/bin/bash** wskazujący położenie interpretera. Następnie poprzez prosty warunek **if**, sprawdzamy czy przy wykonywaniu skryptu podano argument, jeśli nie kończymy programu z kodem wyjściowym **1**. Drugi warunek **if** sprawdza, czy podany w argumencie katalog istnieje. Jeśli nie, program również kończy się z kodem **1**.

Główna część skryptu zaczyna się od ustawienia licznika **i** na wartość **1**. Następnie wykonywane jest stosunkowo długie polecenie **find** łączone potokami, powodujące wykorzystanie wyniku jednego polecenia jako dane wejściowe drugiego polecenia. Rozkładając te długie polecenie uzyskujemy:

- **find “katalog”**: przeszukaj katalog o nazwie **katalog**
- **-maxdepth 1**: przeszukaj tylko katalog główny, nie wchodząc w podkatalogi
- **-type f**: szukaj tylko zwykłe pliki (ignorując katalogi czy linki)
- **-executable**: filtruj wynik, pokazując tylko pliki, do których mamy uprawnienia do wykonywania
- **-printf “%s %p\n”**: wypisz wynik w postaci `<rozmiar pliku w bajtach> <pełna ścieżka do pliku> <znak nowej linii>`
- **| sort -n**: wynik poprzedniego działania posortuj względem rozmiarów pliku (wartości numerycznej)
- **| while read -r size file; do**: dla każdego wiersza z poprzedniego działania przypisuje rozmiar do zmiennej **size** oraz ścieżkę do zmiennej **file**, następnie rozpoczynana się pętla do iterująca przez wszystkie uzyskane pliki.

Wewnątrz pętli wykonujemy prostą podmianę nazwy na nową nazwę zawierającą liczbę określającą jej numer oraz zwiększenie wartości iteratora **i**. Dodatkowo wykonywane zmiany wypisywane są w konsoli przy pomocy **echo** w celu łatwiejszego podglądu działania programu.

```

daniel@DESKTOP-G8BN4D5:~/lab3$ ls -l ./katalog1
total 0
-rwxrwxrwx 1 daniel daniel 6 Nov 12 08:41 plik1
-rw-r--r-- 1 daniel daniel 11 Nov 12 08:42 plik2
-rwxrwxrwx 1 daniel daniel 23 Nov 12 08:42 plik3
-rwxrwxrwx 1 daniel daniel 4 Nov 12 08:42 plik4
daniel@DESKTOP-G8BN4D5:~/lab3$ bash ./zad1.sh katalog1
Zmieniono: katalog1/plik4 -> katalog1/plik4.1
Zmieniono: katalog1/plik1 -> katalog1/plik1.2
Zmieniono: katalog1/plik3 -> katalog1/plik3.3
daniel@DESKTOP-G8BN4D5:~/lab3$ ls -l ./katalog1
total 0
-rwxrwxrwx 1 daniel daniel 6 Nov 12 08:41 plik1.2
-rw-r--r-- 1 daniel daniel 11 Nov 12 08:42 plik2
-rwxrwxrwx 1 daniel daniel 23 Nov 12 08:42 plik3.3
-rwxrwxrwx 1 daniel daniel 4 Nov 12 08:42 plik4.1
daniel@DESKTOP-G8BN4D5:~/lab3$ █

```

Zdjęcie 1: Wynik działania skryptu nr. 1

3. Wnioski

Zadanie zostało poprawnie wykonane, zgodnie z jego treścią. W wyniku tworzenia skryptu zapoznano się w zakresie podstawowym z pojęciem potoku, sposobem łączenia poleceń oraz utrwalono składnię oraz działanie **if** oraz sposobu iterowania przez zawartość folderu lub wynik działania innego polecenia (**do ... done**). Wiedza ta zostanie użyta w celu wykonywania dalszych zadań.

4. Literatura

1. Podręcznik systemowy linux, wywoływany lokalnie poleceniem `man [nazwa szukanego polecenia]`, dostęp online <https://man.he.net> [dostęp: 29.10.2025]. W dniu dostępu strona działała
2. <https://stackoverflow.com/> [dostęp 29.10.2025]. W dniu dostępu strona działała