

**dedSystemy operacyjne 2 Lab - zadania z zajęć.**

## **Lab 1** - lista znana:

[http://www.zsk.ict.pwr.wroc.pl/zsk/repository/dydaktyka/so/so\\_lab1.pdf](http://www.zsk.ict.pwr.wroc.pl/zsk/repository/dydaktyka/so/so_lab1.pdf)

## **Lab 2 <podstawy basha>**

### **Prowadzący: Kamil Szyc**

Zadanie 1: Utworzenie katalogu lab2, w nim utworzenie 4 katalogów, w pierwszym należało stworzyć 3 pliki (wypełnione), w drugim katalogu trzeba było utworzyć dowiązania symboliczne do plików z katalogu pierwszego (do pliku 1 i 2 ze ścieżką względną, do 3 ze ścieżką bezwzględną), trzeci katalog miał zawierać dowiązanie do katalogu 2, w katalogu czwartym miały znaleźć się skopiowane dowiązania z katalogu drugiego. Na początku skryptu należało usunąć katalog lab2- jeżeli istniał.

Zadanie 2: Napisanie skryptu z jednym parametrem, którym był plik (np. z katalogu 1 z poprzedniego zadania). Skrypt ten miał za zadanie znalezienie wszystkich dowiązań do tego pliku w obrębie katalogu lab2.

Zadanie 3: Skrypt tworzący dwa dodatkowe katalogi (5 i 6, chyba jako parametry), dodający do nich pliki, a następnie tworzący w katalogu 5 dowiązania do plików z katalogu 6 i odwrotnie. Możliwe, że pliki i katalogi można było tworzyć ręcznie, a sam skrypt miał się skupiać tylko na tworzeniu dowiązań (Jednak radzę utworzyć je w skrypcie tak jak w zadaniu 1). Mówił też coś o sprawdzeniu poprawności podanych ścieżek w parametrach.

PS. Początkowo zadanie pierwsze było na 3, drugie na 4, a trzecie na 5, jednak jak zobaczył, że po godzinie wszyscy siedzą nad pierwszym dalej, to zmienił zasady. Pierwsze na 4 a jedno z tych dwóch na 5.

### **Prowadzący: Doktor Jarosław Rudy**

Wspólne: Wypisać na ekran wszystkie linki symboliczne z danego katalogu (1-szy argument skryptu), które wskazują na zadane miejsce (ścieżkę) w systemie plików (2-gi argument). Zadane miejsce może znajdować się poza zadanym katalogiem

### **Prowadzący: Szymon Datko**

//losujemy 1 zadanie, jak ktoś zrobi to 2 na wyższą ocenę

### **Zadania podstawowe:**

21. Napisz skrypt, który dla wszystkich plików z jakiegos katalogu (parametr wywołania skryptu) utworzy dowiązania symboliczne w drugim katalogu (parametr wywołania skryptu), jeśli odpowiednie dowiązania symboliczne jeszcze nie istnieją.

22. Napisz skrypt, który dla wszystkich plików wykonywalnych z jakiegoś katalogu (parametr wywołania skryptu) utworzy dowiązania twarde w drugim katalogu (parametr wywołania skryptu), zaś dla podkatalogów utworzy on dowiązania miękkie.

25. Napisz skrypt, który wyświetli nazwy wszystkich dowiązań symbolicznych z danego katalogu (parametr wywołania skryptu), które wskazują na określone miejsce w systemie plików (parametr wywołania skryptu).

**Prowadzący: Paweł Lampe**

Pierwsze trzeba było zrobić zadanie, które polegało na stworzeniu kilku folderów, w jednym z nich były dwa pliki i skrypt, a głębiej trzeba było utworzyć dowiązanie do skryptu. Skrypt miał wypisać ścieżkę skryptu, zawartość tych dwóch plików i miał być odpalony przez dowiązanie.

Później prowadzący pokazał nam jak oddzielić errorry od coutów w c++.

**Prowadzący: Mariusz Uchroński**

**Prowadzący: Cariusz Daban**

W zadanym katalogu przerobić dowiązania twarde do plików regularnych w tym samym katalogu na dowiązania symboliczne.

## Lab 3 <find>

**Prowadzący: Kamil Szyc**

Zadanie 1

Struktura plików do utworzenia w skrypcie.

Należy znaleźć pliki o rozszerzeniu txt do których właściciel ma prawo wykonywania, ale nie czytania

Znaleźć pliki txt lub mp3 których rozmiar jest większy od zera (a nie różny od zera) i które są plikami

Zadanie 2

Tworzymy katalogi

kat1

kat2/kat3

kat2/kat4tes

teksty obrazy muzyka inne

generujemy 300 plikow w petli w nastepujacy sposob:

jeżeli  $i \% 3 == 0$

tekst.txt -> kat1, tekst.txt -> kat2/kat3

jeżeli  $i \% 5 == 0$

obraz.jpg -> kat1, obraz.jpg -> kat2/kat4

jeżeli  $i \% 7 == 0$

muzyka.mp3 -> kat2/kat3, muzyka.mp3 -> kat2/kat4

else

plik.dat -> kat1, plik.test -> kat1

Następnie przenosimy poleceniem cp pliki .txt do teksty, .jpg do obrazy itd, a kat1 i kat2 usuwamy

**Prowadzący: Jarosław Rudy**

**Prowadzący: Szymon Datko**

**Podstawowe:**

43. (?) Napisz skrypt, który w zadanym miejscu systemu plików (parametr wywołania skryptu) o ile ono istnieje, wyświetli ścieżki do wszystkich plików (także w podkatalogach) które są wykonywalne dla wskazanego użytkownika (parametr wywołania skryptu)

44. Napisz skrypt, który wyświetli ścieżki do wszystkich lokalnych kopii, nie starszych niż miesiąc, repozytoriów kodu (katalogów, zawierających podkatalog o nazwie '.git'), znajdujących się w określonym miejscu systemu plików (parametr wywołania skryptu), o ile ono istnieje.

45. Napisz skrypt, który dla podanego miejsca w systemie plików (parametr wykonania skryptu) oraz wszystkich jego podkatalogów, wypisze ścieżkę, nazwę użytkownika, grupy i uprawnienia do plików zwykłych, których formalnym właścicielem nie jest aktualny użytkownik (whoami)

**Rozszerzone:**

54. Napisz skrypt, który będzie pełnił funkcję programu czyszczącego katalog użytkownika ze śmieci

- a) program powinien zliczyć wszystkie puste pliki i katalogi w zadanym miejscu (parametr wywołania skryptu), należące do danego użytkownika (parametr wywołania skryptu)
- b) program powinien zgłosić pliki i katalogi, których zawartość nie była zmieniana od roku
- c) program powinien zaraportować wszystkie pliki, których zadany użytkownik nie jest oficjalnym właścicielem (nie uwzględniamy grupy użytkownika)
- d) program powinien zgłosić wszystkie pliki, których wskazany użytkownik nie może przeczytać

**Prowadzący: Paweł Lampe**

**Prowadzący: Mariusz Uchroński**

zad. Wyświetlic pliki z podziałem na trzy grupy: 222, 750, !-077

zad. Podać zadaną liczbę pozycji(argument \$2) ostatnio modyfikowanych oraz najdawniej modyfikowanych z katalogu(argument \$1))

**Prowadzący: Dariusz Caban**

W zadanym drzewie katalogów znaleźć pliki, mające dowiązania twarde do których właściciel ma prawo odczytu i zapisu, a nie ma prawa wykonania, natomiast grupa lub inni mają prawo wykonania.

## Lab 4 <while, read, potoki>

**Prowadzący: Kamil Szyc**

zad1

Wyświetlenie wszystkich ścieżek kanonicznych znajdujących się we wszystkich podkatalogach, wyświetlenie wielkości plików (tych plików które znajdziemy)  
Całość wyszukiwań ograniczyć do 3 plików. Pliki >0. Wszystkie / na \ w wypisanej ścieżce.  
(komenda tr)

zad2

a) Wyświetlenie informacji ile plików jest w każdym podkatalogu. Realfiles i nazwę podkatalogu. " W tym katalogu jest np.5 plików" Zapisać na ekranie i do pliku. Posortować malejąco.

b)

#Skorzystać z: find, while, read - wczytywanie danych, sort, tee - wyświetla na ekranie i interpretuje do pliku, wc -l - zliczanie plików , tr - podmiana ukośników, head - wyświetlenie np. 3 elementów

**Prowadzący: Jarosław Rudy**

**Prowadzący: Szymon Datko**

**Podstawowe:**

61. Napisz skrypt, który zaszyfruje (szyfr cezara; program tr) każdy plik obecny w podanym miejscu systemu plików (parametr wywołania skryptu). Uwzględnij wszystkie możliwe podkatalogi wskazanego miejsca oraz upewnij się, że podmiot uruchamiający skrypt ma odpowiednie prawa do szyfrowanych plików. Wyświetl na ekranie i zapisz do pliku ścieżki do szyfrowanych plików

(62) Napisz skrypt, który przeszuka zadane miejsce w systemie plików (parametr wywołania skryptu), a następnie wyświetli ścieżki do wszystkich plików z tego miejsca (uwzględniając wszystkie możliwe podkatalogi) w formacie typowym dla systemów z rodziny DOS i Windows - to znaczy ze znakiem \ zamiast / oraz dowolnie wybranym oznaczeniem partycji (może być po prostu C:).

63. Napisz skrypt, który z zadnego miejsca systemu plików (parametr wywołania skryptu), uwzględniając wszystkie możliwe podkatalogi wypisze ścieżki do N (opcjonalny parametr

wywołania skryptu; domyślnie 10) największych (pod względem rozmiaru w bajtach) plików. Jeżeli kilka plików ma taki sam rozmiar, wybieramy dowolny z nich (bez znaczenia który).

#### **Rozszerzone:**

(74) Zabawy z komenda netstat -atunp. Napisz skrypt, który:

- a) policzy, ile jest wszystkich otwartych portów sieciowych; ile z nich dotyczy połączeń przychodzących (wartość LISTEN w polu State)
- b) wypisze ile ze wszystkich otwartych portów jest skojarzonych z różnymi protokołami (tcp, ...)
- c) zamieni wielokrotne wystąpienia spacji na pojedyncze ich wystąpienia
- d) wypisze wszystkie numery otwartych portów na lokalnej maszynie, szczególnie wskazując te, które najprawdopodobniej skojarzone są z jakimiś lokalnymi serwerami (wartość LISTEN ...)

**TIP:** <http://www.thegeekstuff.com/2010/03/netstat-command-examples>

**Prowadzący: Mariusz Uchroński**

#### **Podstawowe:**

Zadane dwa katalogi (drzewa katalogów) jako parametr. Należy wpisać pełną ścieżkę do plików które są w pierwszym folderze ale nie ma ich w drugim i na odwrót. (Tzw unikalne).

**Podstawowe v2:** Dane są dwie ścieżki, przeszukać je za pomocą find. Używając w potoku while znaleźć pliki które się powtarzają ale mają inne rozszerzenia np. nazwa.txt i nazwa.wav. Wypisane linijki mają się nie powtarzać.

#### **Dodatkowe:**

Zad1. Zadane dwa katalogi (drzewa katalogów) jako parametr. Należy wypisać w formacie <ściezka do katalogu1> <ściezka do katalogu2> ściezki do plików o tej samej nazwie ale różnych typach (np plik, katalog, dowiązanie) znajdujących się w tych katalogach zadanych jako parametr.

**Prowadzący: Paweł Lampe**

**Prowadzący: Dariusz Caban**

Wyswietlic pliki regularne w drzewie o tej samej nazwie wystepujace conajmniej n razy (n jako drugi paramter)

## Lab 5 <Potokowe przetwarzanie strumieni tekstowych (grep, awk)>

**Prowadzący: Kamil Szyc**

**Zad**

działania na pliku /etc/passwd

\*"Użytkownik xxx, którego katalog domowy to xxx, używa powłoki ..."

/bin/false oraz /user/sbin/nologin - nie wypisywać

\*Wyświetlenie wszystkich grup, które pojawiają się częściej niż raz

\*Users xxx,xxx, have x group

\*Users xxx,xxx have x group

^ostatnie chodzi o to, żeby wypisać users bez tego przecinka na końcu

Druza grupa:

Zajmujemy się ps aux (polecenie) - polecenia awk

Wyświetlić komunikaty:

a) Użytkownik xxx ma uruchomiony proces xxx (CPU = xxx, MEM = xxx), tylko te wartości, gdzie procesor lub pamięć ma zużycie > 0.1

b) To samo tylko, proces (ze ścieżki ../../../../) zamienić na samą ostatnią część

c) Podliczyć całkowite zużycie procesora i pamięci

CPU = xxx

MEM = xxx

d) Jak powyżej jednak dla konkretnego użytkownika (Wylistować wszystkich użytkowników)

Użytkownik xxx używa CPU = xxx, MEM = xxx

e) Wyświetlić, który użytkownik zużywa najwięcej procesora oraz ten, który zużywa najwięcej pamięci

**Prowadzący: Jarosław Rudy**

**Prowadzący: Szymon Datko**

**Podstawowe:**

81. Napisz skrypt, który przeanalizuje listę wszystkich dostępnych użytkowników w systemie (czyli zawartość pliku /etc/passwd) i wypisze nazwy tych użytkowników (pierwsze pole w pliku), którzy mają ustawiony katalog główny (czyli /) lub czarną dziurę (czyli /dev/null) jako katalog domowy (przedostatnie pole w pliku).

82. Napisz skrypt, który przeanalizuje listę wszystkich dostępnych użytkowników w systemie (czyli zawartość pliku `/etc/passwd`) i wypisze nazwy tych użytkowników (pierwsze pole w pliku), mających ustawioną wartość `/bin/false`, `/usr/bin/nologin` lub `/sbin/nologin` jako domyślną powłokę (ostatnie pole w pliku).

83. Napisz skrypt, który będzie pełnił funkcję bardzo prostego analizatora składni plików csv. Załóżmy optymistycznie, że każdy poprawny plik csv składa się z nagłówka w pierwszej linii, zawierającego spis pól w pliku (na przykład: `#id,imie,nazwisko` - zgłoś (wypisz) linie, które nie zawierają odpowiedniej liczby pól

(86)Napisz skrypt, który będzie obliczał sumaryczne obciążenie procesora oraz zajętość pamięci operacyjnej, analizując wynik działania komendy `ps aux` (odpowiednie dane znajdujące się w trzeciej oraz czwartej kolumnie wyniku). [poprosze o jedno, zmyslne wykonanie komendy `awk`]

### **Rozszerzone:**

(91 - {nie pamiętam dokładnie treści})

Napisać skrypt, który:

- a) w podanej jako parametr lokalizacji znajdzie pliki charakterystyczne dla języka C
- b) wskaże pliki i linie w których znajdują się odwołania do niebezpiecznych funkcji (np. `gets`, `strcpy`, `sprintf`)
- c) sprawdzi czy pliki załączone w `#include` istnieją w `/usr/include` lub w katalogu w którym znajduje się plik
- d) znajdzie w plikach zadeklarowane zmienne, które nie zostały użyte (w ramach najbliższego bloku `{ }`)

(92)Proszę napisać konwerter plików z formatu json do formatu xml. Niech skrypt:

- a) wczyta wskazany plik json (parametr wywołania skryptu) i zapisze wynik obok niego, w pliku o odpowiednim rozszerzeniu; proszę skontrolować także poprawność argumentów
- b) dokona konwersji, traktując klucze jako nazwy znaczników, czyli `{klucz: wartosc}` stanie się `<klucz>wartosc</klucz>`; zakładamy, iż dany plik json ma poprawną składnię
- c) wyświetli samo-zamykający się znacznik (`<klucz />`), jeśli wartość jakiegoś pola to pusty ciąg (`"` lub `""` typowo) lub w ogóle brakuje wartości
- d) spróbuj napisać skrypt tak, aby wspierał on zagnieżdżanie wartości (czyli dopuszczamy format `{klucz1: wartosc, klucz2: {podklucz1: cos1, podklucz2: cos2}}`)

95. Napisz pozostawiony markerem w toalecie głosił: "dokument tajemnic został otwarty -- strzeżcie się wrogowie dziedzica". Uratuj świat, używając `awk` jako swojego zaklęcia Patronusa i zabawiając się z <http://tinyurl.com/doc-tajemnic>. Niech skrypt:

- a) wyciągnie zawartość dokumentu z podanego adresu sieciowego, używając na przykład komendy `wget -qO- <adres>`
- b) poda, ile jest unikalnych prowadzących w tym dokumencie, przyjmując naiwnie, iż wpisy 'Doktor Jarosław Rudy' i 'Jarosław Rudy' to różni prowadzący



- c) oszacuje objętość zadań (liczoną na przykład jako liczbę znaków, zgłoszonych dla każdego prowadzącego i wyświetli zestawienie według popularności wpisów
- d) wyszukując odpowiednie wzorce (wiedząc na przykład, że zadania Pana Szymon Datki stosują zawsze numerację dwucyfrową) wyznaczy dokładną liczbę zadań dla każdego prowadzącego

**Prowadzący: Mariusz Uchroński**

**Prowadzący: Paweł Lampe**~~~~~

**Prowadzący: Dariusz Caban**

## Lab 6

**Prowadzący: Szymon Datko**

Zadania ze strony [regexone.com](http://regexone.com)

Jest 15 zadań tutorialowych (minimum 10 na ocenę 3.0) i 5 poleceń. Za każde polecenie 0.5 oceny w górę. Dodatkowo pod każdym zadaniem jest solution i co prawda Szymon nie lubi, jak ktoś od razu wchodzi do solution i przepisuje wynik, ale zawsze można popatrzeć, jak to powinno wyglądać i spróbować samemu coś napisać. Generalnie polecam ^^

^ te zadania tylko na stronie czy skrypty w bashu normalnie piszemy? Te 5 poleceń też są na stronie ????

Tak, all na stronie. Szymon chodzi i patrzy, jak tam każdemu idzie. Te pierwsze 15 tutorialowych zadań można mu nie pokazywać (albo pokazać jakąś część). Co do 5 poleceń to trzeba je krótko omówić (Szymon sprawdza, czy zrobiłeś sam, czy spisałeś), ale nie robi wielkiej spiny, nawet jak nie zrobiło się 100% dobrze.

**Prowadzący: Kamil Szyc**

3 pierwsze zadania to były wyrażenia regularne. Za pomocą strony <http://regex101.com> trzeba było stworzyć 3 wyrażenia regularne dla: godziny z użyciem PM i AM, dla daty w wybranym przez siebie formacie np DD.MM.YYYY albo DD-MM-YYYY z tym że dla YYYY trzeba było zapisać jeszcze jakiś warunek, i dla strony www.

Później były zadanka, chyba 3 lub więcej.

1. Polegało na tym żeby ściągnąć z jakiejś bazy pierdyliard adresów mailowych i teraz należało do adresu dopisać czy jest poprawny czy nie za pomocą słowa TAK, NIE. Niepoprawny adres był najczęściej bez @.
  2. Należało zamienić BardzoDługiTekst -> bardzo\_dlugi\_tekst. Tyle pamiętam.
- Pan Kamil zadawał też pytanko, co tu robi \* lub :\ itd itp :D można było korzystać z internetów ale świadomie a nie na pałę kopiować.

## Lab 7

**Prowadzący: Kamil Szyc**

Zad1: plik zawierający listę plików wiersz po wierszu- jako parametr wyświetlenie na ekran oraz zapisanie do nowego pliku nazwy plików wraz z ich treścią.

Zad2: katalog A w którym są pliki, B w którym są pliki. Napisanie skryptu przyjmującego jako parametry katalogi. Utworzyć linki symboliczne w katalogu A do plików z katalogu B i odwrotnie.

**Prowadzący: Szymon Datko**

### **Podstawowe:**

(A2) Napisz skrypt, który wyświetli na ekranie połączona zawartość wszystkich plików, wskazanych na liście, znajdującej się w pliku ze spisem (parametr wywołania skryptu). Zakładamy, że każda linia spisu to osobna ścieżka oraz, że wszystkie pliki istnieją. Łączymy w kolejności ze spisu. [tylko Perl]

**blagam, niech ktos napisze bo trudne :/**

**Proszę:**

- otwórz spis, w pętli czytaj linie; każda linia (pamiętaj o chomp) to nazwa pliku - otwórz plik, w pętli czytaj i wypisuj całą jego zawartość, zamknij plik; koniec
- <http://www.zsk.ict.pwr.wroc.pl/zsk/repository/dydaktyka/so/instrukcje/perl1b.pdf> strony 6, 7, 5 (w tej kolejności)

Dzięki bardzo, dużo to pomogło, udało się zdać laborkę, a przy okazji coś się nauczyłem! :)

### **Rozszerzone:**

(B1) Proszę opracować proste narzędzie, które będzie Twoją własną wariacją narzędzia head - czyli powinien on wyświetlać kilka pierwszych linii zawartości wskazanego pliku (parametr wywołania skryptu). Dodatkowo niech skrypt:

- a) skontroluje, czy plik podany mu jako argument, na pewno istnieje
- b) sprawdzi, czy wskazany plik jest plikiem zwykłym i czy można go przeczytać
- c) umożliwi określenie, ile początkowych linii z pliku należy wyświetlić, jeśli podana zostanie flaga -n jako pierwszy argument, a zaraz za nią (jako kolejny argument) zadana liczba
- d) obsługuje możliwość podania kilku plików do przeczytania - należy wtedy zawartości każdego z nich poprzedzić nagłówkiem z nazwą pliku [punkty a) i b) także powinny być uwzględnione]

## Lab 8

**Prowadzący: Szymon Datko**

A1: Wyszukaj pliki w danym katalogu (parametr wywołania skryptu) oraz wszystkich jego podkatalogach, które są dowiązaniem symbolicznym do pliku zwykłego, który można przeczytać, i czas modyfikacji nie jest większy niż parę minut. Żadnych innych warunków nie trzeba sprawdzać.

B3: Proszę opracować narzędzie, które będzie zaspokajało obsesyjno-kompulsywne potrzeby analiz zawartości danego miejsca w systemie plików (parametr wywołania skryptu), czyli katalogu roboczego ze wszystkimi jego podkatalogami. Niech skrypt:

- a) skontroluje liczbę argumentów oraz istnienie i stosowne uprawnienia do wskazanego katalogu
- b) wyznaczy liczbę plików na każdym poziomie zadanego drzewa podkatalogów (przykładowo najpierw katalog a/, potem a/b i a/e, a później a/b/c, a/b/d, a/e/f)
- c) zliczy ile w danym miejscu jest plików o różnych rozszerzeniach, w tym także bez rozszerzenia, zaś dla dowiązań miękkich poda ile z nich wskazuje ścieżkę względną, a ile bezwzględną
- d) wspiera możliwość podania więcej niż jednego katalogu roboczego -- należy wtedy wykonać analizę każdego katalogu osobno, na koniec wyświetlić zbiorcze zestawienie