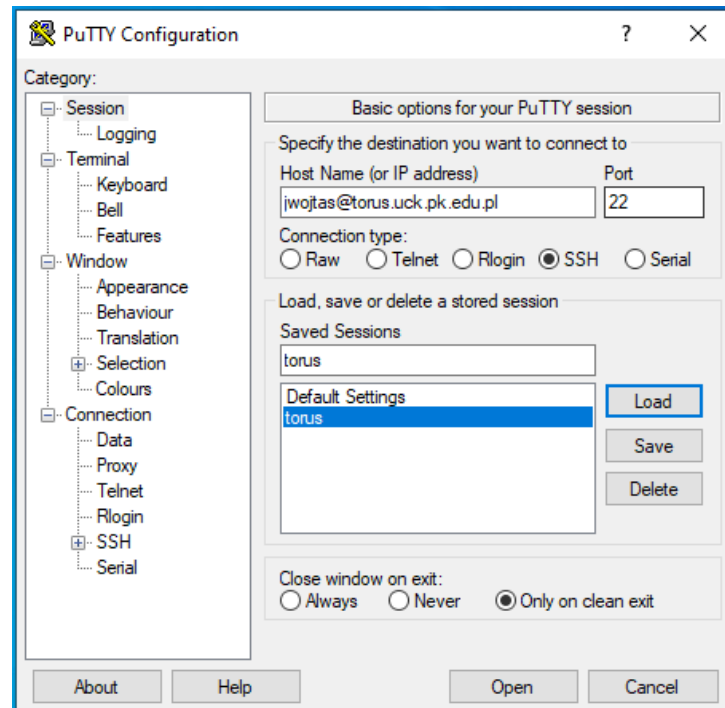


Systemy operacyjne – studia stacjonarne 2023/24

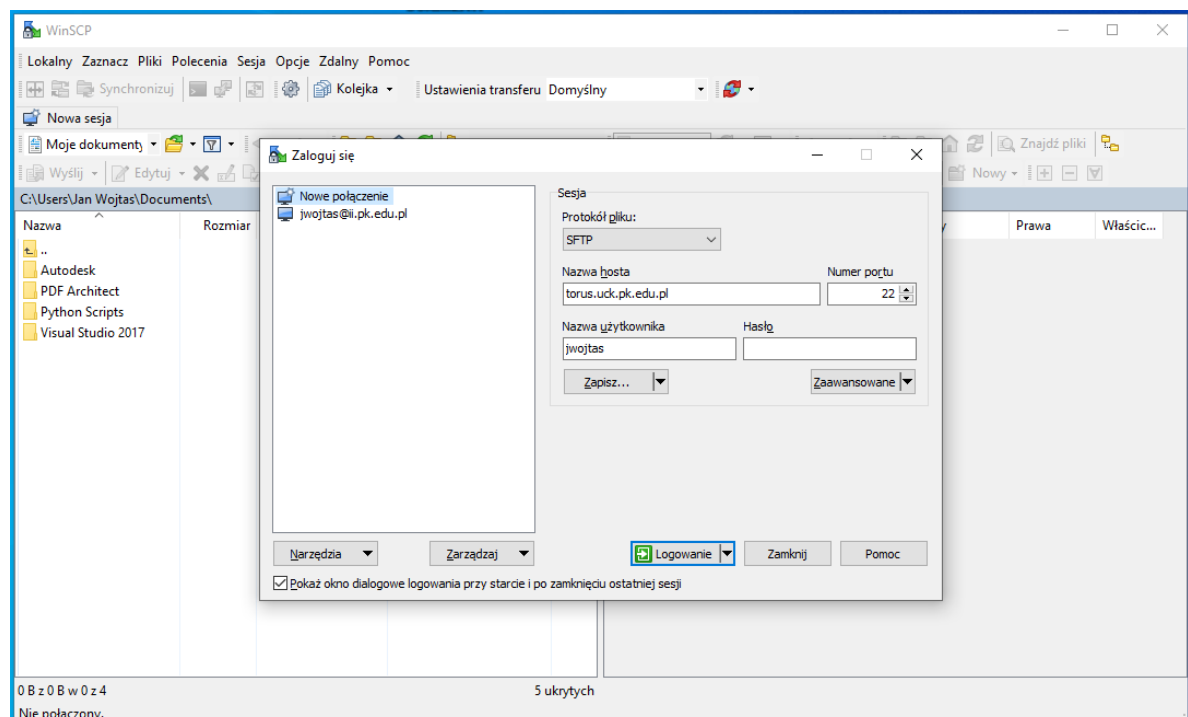
Lab 1. Omówienie tematyki zajęć, warunki zaliczenia. Polecenia systemu Linux. Standardowe wejście wyjście, przekierowania, potoki.

1. Praca z systemu Windows. Logowanie (PuTTY) na serwer: **torus.uck.pk.edu.pl**
IP: 149.156.136.154



Login i hasło zostaną podane podczas laboratorium. Po zalogowaniu należy zmienić hasło poleceniem **changeMyPwd** (dawniej **passwd**)

Przesyłanie plików (WinSCP) na serwer: **torus.uck.pk.edu.pl**



2. Praca z systemu Linux.

Logowanie (ssh) na serwer: **torus.uck.pk.edu.pl**

```
[jwojtas@ii ~]$ ssh jwojtas@torus.uck.pk.edu.pl
jwojtas@torus.uck.pk.edu.pl's password: █
```

Przesyłanie plików (scp) na serwer: **torus.uck.pk.edu.pl**

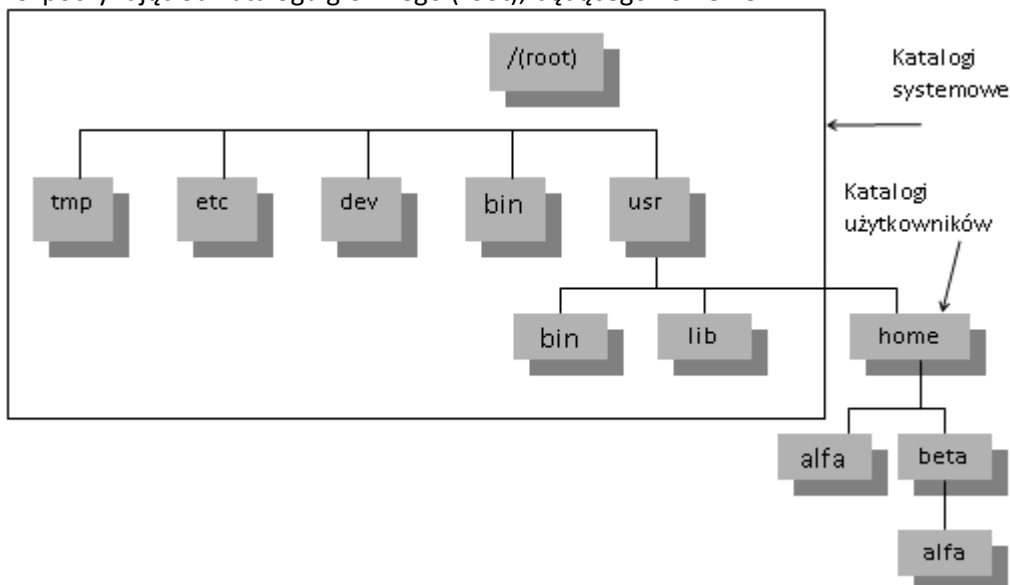
```
[jwojtas@ii ~]$ scp plik.txt jwojtas@torus.uck.pk.edu.pl:plik.txt █
```

3. Podstawowe polecenia Linux-a (przypomnienie z WDP) – grupy poleceń.

- | | |
|--|---|
| • cd, pwd, ls, tree, who, finger, id, ... | przeglądanie struktury plików i katalogów, informacje o użytkownikach |
| • cat, more, less, head, tail, ... | przeglądanie zawartości plików |
| • cp, mv, rm, mkdir, rmdir, touch, ... | modyfikacja struktury plików i katalogów |
| • nano, pico, vi, vim, mcedit, gvim, emacs, ... | edytory |
| • wc, nl, sort, cut, grep, find, which, cal, ... | operacje na strumieniach/plikach |
| • ps, pstree, top, jobs, fg, bg, kill, ... | informacje o procesach |
| • man, info, help | bazy pomocy |

4. Charakterystyka systemu plików.

W Linux-ie pliki są umieszczone w katalogach, podobnie jak ma to miejsce w DOS-ie/Windows-ie. Katalogi z kolei są ze sobą hierarchicznie powiązane w jedną strukturę plików. Nazwa pliku może składać się z liter, cyfr oraz niektórych znaków specjalnych, np. znaku podkreślenia. Jej długość może wynosić do 256 znaków. Linux rozróżnia duże i małe litery. Kropka nie posiada specjalnego znaczenia, ponieważ Linux traktuje kropkę jak każdy inny znak. Pliki, których nazwy rozpoczynają się kropką są „ukryte”. Traktowane są jak wszystkie inne pliki, z wyjątkiem tego, iż nie wyświetli ich polecenie ls, chyba że użytkownik zastosuje opcję `-a`. Wszystkie pliki mają jeden format fizyczny – ciąg bajtów zakończony znakiem EOF (End-Of-File; Ctrl-D). Ten hierarchiczny system plików jednolicie traktuje pliki zwykłe, katalogi oraz pliki opisujące urządzenia zewnętrzne. Katalogi każdego użytkownika są w rzeczywistości połączone z katalogami pozostałych użytkowników. Są one ułożone w strukturę hierarchiczną drzewa, rozpoczynając od katalogu głównego (root), będącego korzeniem.



Przy rejestrowaniu nazwy użytkownika w systemie zostaje z nią związany katalog osobisty danego użytkownika, oznaczany symbolem ~. Katalog ten staje się katalogiem bieżącym w chwili rozpoczęcia przez użytkownika sesji przy terminalu. Każda nazwa pliku, którą podaje wówczas użytkownik, np. jako parametr polecenia, odnosi się do plików z katalogu bieżącego. Każdy plik można znaleźć rozpoczynając poszukiwanie od katalogu pierwotnego, tj. od korzenia drzewa katalogów.

W każdym katalogu występuje element domyślny, stanowiący nazwę katalogu nadrzędnego, oznaczony dwiema kropkami (..), oraz inny element, stanowiący nazwę katalogu, w którym on sam się znajduje, oznaczony (.). Innymi słowy, każdy katalog wskazuje na samego siebie oraz na swój katalog nadrzędny.

5. Znaki specjalne nazw plików: *, ?, [].

Nazwy plików są najczęściej argumentami stosowanymi w poleceniach. Powłoka oferuje zestaw znaków specjalnych, które poszukują, dopasowują i generują listę plików. Tymi znakami specjalnymi są gwiazdka, znak zapytania i nawiasy kwadratowe. Znaki * i ? określają niepełną nazwę pliku, nawiasy kwadratowe pozwalają na określenie zestawu dopuszczalnych znaków, jakie mają być poszukiwane. Można łączyć nawiasy kwadratowe z innymi znakami specjalnymi.

- **Gwiazdka *** oznacza dowolny ciąg znaków, można jej używać do oznaczania nazw plików zaczynających się lub kończących dowolnym zestawem znaków.
- **Znak zapytania ?** oznacza dokładnie jeden dowolny znak. Można używać więcej niż jednego znaku zapytania w każdym miejscu wzorca.
- Dowolne znaki umieszczone w **nawiasach kwadratowych []** oznaczają dokładnie jeden spośród wymienionych znaków.

Znaki specjalne mogą być maskowane za pomocą znaku \, umieszczenie ich w pojedynczych ' ' lub podwójnych " " apostrofach.

6. Standardowe wejście/wyjście, przekierowania, potoki.

Wszystkie pliki Linux-a są logicznie zorganizowane jako ciągły strumień bajtów. Fizycznie natomiast pliki są ułożone jako rozrzucone bloki w pamięci dyskowej. Poza wywołaniami systemowymi użytkownik nigdy nie odwołuje się do fizycznej struktury pliku. Dla użytkownika wszystkie pliki mają identyczną organizację – strumienia bajtów. Każdemu procesowi w chwili uruchomienia przypisane są automatycznie standardowe strumienie: wejściowy (stdin), wyjściowy (stdout) i wyjściowy błędów (stderr). Dane wejściowe z klawiatury są umieszczane w strumieniu danych jako ciąg bajtów. Dane wyprowadzane z polecenia czy programu są również umieszczane w strumieniu danych w postaci ciągu bajtów. Wejściowy ciąg danych jest w Linux-ie nazwany standardowym wejściem (KLAWIATURA), a wyjściowy (EKRAŃ) – standardowym wyjściem.

Standardowe wejście lub wyjście użytkownik może zmienić dzięki przekierowaniom, np. można skierować standardowe wyjście do pliku, a nie na ekran. Aby to było możliwe, należy umieścić operator przekierowania (>, <, >>) i nazwę pliku w wierszu poleceń.

ls > alfa

Polecenie ls wyświetla na ekranie pliki znajdujące się w bieżącym katalogu. Standardowe wyjście można przekierować do pliku np. ls > alfa, wówczas wynik polecenia ls zamiast na ekranie znajdzie się w pliku o nazwie alfa. Jeśli plik o nazwie alfa (docelowy) istnieje, to zostaje nadpisany, jeśli nie istnieje, to zostanie utworzony.

Do przekierowania można także używać operatora >>. Polecenie ls >> alfa powoduje dopisanie do pliku alfa (gdy plik o takiej nazwie istnieje) wyniku polecenia ls. Jeśli plik alfa nie istnieje, to zostanie utworzony. Operator przekierowania jest wykonywany wcześniej niż zasadnicza komenda.

Do standardowych operacji strumieni można odwołać się przez ich numery:

- 0** – standardowe wejście (klawiatura) (/dev/stdin),
- 1** – standardowe wyjście (monitor) (/dev/stdout),
- 2** – standardowe wyjście błędów (monitor) (/dev/stderr).

Przykład: **cat plik 1>alfa 2>beta**

W powłoce BASH można odwołać się do strumienia standardowego przez poprzedzenie znakiem **&** jego numeru, np. **&1** odnosi się do standardowego wyjścia. Operacja **2>&1** przekierowuje standardowe wyjście błędów na standardowe wyjście.

Uwaga: /dev/stdin, /dev/stdout, /dev/stderr, /dev/null, /dev/zero to urządzenia systemowe. Znasz inne urządzenia tego typu?.

Ćwiczenie 1.

- a) Wykonaj komendę **ls -l**.
- b) Sprawdź czy plik o nazwie **dane** istnieje, jeśli tak, usuń istniejący plik.
- c) Wykonaj komendę **ls -l > dane**.
- d) Sprawdź jaka jest zawartość pliku **dane**: czy jest tam zawarty również opis pliku **dane**?
- e) Jaki był rozmiar pliku **dane** przed wykonaniem komendy **ls -l** (odczytaj z pliku **dane**)?
- f) Jaka jest teraz zawartość i wielkość pliku **dane**?
- g) Wykonaj komendę **ls -i >> dane**.
- h) Sprawdź jaka jest zawartość pliku **dane**.
- i) Usuń plik **dane**.
- j) Wykonaj komendę **ls -i >> dane**.
- k) Sprawdź jaka jest zawartość pliku **dane**: czy jest tam zawarty również opis pliku **dane**?
- l) Wykonaj komendę **cat dane > dane**.
- m) Co się stało z plikiem **dane**? Sprawdź jaki jest rozmiar i zawartość pliku **dane**.
- n) Wyjaśnij dlaczego tak się stało.

Potoki. Mogą mieć miejsce takie sytuacje, w których chcemy przesłać standardowe wyjście do innego polecenia, a nie do pliku wynikowego. Aby takie połączenie wykonać, trzeba wykorzystać tzw. **potok**. Operator potoku **|** (pionowa kreska), umieszczony między poleceniami, tworzy połączenie między nimi:



cat plik.txt | sort

Gdy chcemy przekierować standardowe wyjście do pliku, a równocześnie wyświetlić wyprowadzoną zawartość na ekranie, musimy skorzystać z polecenia **tee**, które rozgałęzia strumień wyjściowy. Polecenie **tee** kopiuje standardowe wyjście do pliku.

cat plik.txt | sort | tee x | wc -l

Ćwiczenie 2.

- a) Utwórz plik o nazwie **proba**, składający się przynajmniej z 5 linii tekstu.
- b) Wyświetl posortowaną zawartość pliku **proba** na ekranie i zapisz do pliku **prsort**.

- c) Wyświetl zawartość katalogu nadrzędnego względem katalogu domowego (~). Strumień wyjściowy tego polecenia skieruj na wejście polecenia **more**. Co otrzymasz na monitorze?
- d) Policz ilość znaków, słów oraz linii, jakie zostaną wyświetlone na monitorze po wykonaniu polecenia **who**.
- e) Zawartość pliku **proba** wyświetl na ekranie oraz skopiuj do pliku **gamma**.
- f) Do pliku **pl1** przekieruj polecenie **ls -i**, natomiast do pliku **pl2** polecenie **ls -s**.
- g) Zapisz w pliku **wynik** posortowane połączone pliki **pl1** oraz **pl2** z numeracją wierszy.
- h) Zapisz w pliku **wynik** oraz wyświetl na ekranie posortowane pliki **pl1** oraz **pl2** z dołączoną numeracją wierszy.
- i) Używając przekierowania dołącz plik **pl1** do pliku **pl2**.
- j) Zaproponuj polecenie tworzące plik **pl3**, będący połączeniem plików **pl1** i **pl2**.

*Treści oznaczone kursywą pochodzą z różnych źródeł internetowych.