

# Podstawy Programowania

## Wykład V

### *Wprowadzenie do systemu X Window. Praca z uniksowym interpreterem poleceń*

*Robert Muszyński  
ZPCiR IIAiR PWr*

**Zagadnienia:** Założenia i historia, model konstrukcji, architektura, serwer i klienty X Window, zarządzca okien, praca z systemem X Window, autoryzacja, zasoby, polecenia powłoki, potoki i przekierowania, aliasy, skrypty.

Copyright © 2010–2015 Robert Muszyński

---

Niniejszy dokument zawiera materiały do wykładu na temat podstaw programowania w językach wysokiego poziomu. Jest on udostępniony pod warunkiem wykorzystania wyłącznie do własnych, prywatnych potrzeb i może być kopiowany wyłącznie w całości, razem ze stroną tytułową.

## Zawartość wykładu

Obecnie wykład ten bazuje na dwóch wykładach kursu [Kurs pracy w systemie Linux](#), do których odnośniki podano w kolumnach **folie**, **wydruki** tabeli z materiałami do wykładów, i które po części zostaną wykorzystane w trakcie zajęć. Ponieważ na wykładzie pojawią się przede wszystkim przykłady ilustrujące opisywane w dostarczonych materiałach mechanizmy, zaleca się wstępne, samodzielne zapoznanie z tymi materiałami.

Na kolejnych slajdach, dla ułatwienia analizy prezentowanych na wykładzie przykładów, dotyczących pracy z uniksowym interpreterem poleceń, oraz by zachęcić do samodzielnego ich modyfikowania, zebrano ich finalne postacie.

Materiały internetowe:

- <http://askubuntu.com/questions/65083/what-kinds-of-desktop...> – zestawienie dostępnych środowisk graficznych systemu X Window
- <http://kurslinux.ovh.org/> – podstawy programowania skryptów (niestety obecnie strona ma „zepsute” menu – jednakże wystarczy pojawiający się w nim adres <http://www.majerowski.pl> zamienić na <http://kurslinux.ovh.org>)
- <http://andyk.pg.gda.pl/skrypty/> – zestaw prostych skryptów

## Przykładowe polecenia i skrypty

- analiza zajętości dysku – informacje o plikach
  - ★ skrypt podsumowujący informacje dla bieżącego katalogu – plik `licz.sh` zawarty w kolumnie **materiały** tabeli z materiałami na laboratorium,
  - ★ uzyskiwanie informacji o wielkości katalogów
    - `alias lhs='du -sh *|sort -h'`
- poszukiwanie/przetwarzanie znalezionych plików
  - ★ znalezienie najnowszej wersji pliku o podanej nazwie
    - `find . -name kubus.txt | xargs ls -lt | head -n1`
  - ★ zmiana rozszerzenia pliku
    - `for file in *jpeg; do`  
`mv $file 'basename $file .jpeg'.jpg; done`
  - ★ zmiana nazw plików z dużych liter na małe – skrypt `dnm.sh` zawarty w kolumnie **materiały** tabeli z materiałami na laboratorium,

- ★ przygotowanie miniaturki pierwszych stron plików pdf

```
find . -name "*.pdf" -exec
    convert -thumbnail x80 {}[0] {}-thumb.png \;
for i in *.pdf; do
    convert -thumbnail x80 $i[0] $i-thumb.png; done
```

- Inne

- ★ sprawdzenie do jakich grup należą obecnie zalogowani użytkownicy

```
for i in `finger |cut -f1 -d ' '`; do
    groups $i; done|sort |uniq -c
```

- ★ okresowe sprawdzanie czy jest ktoś zalogowany – skrypt `czy_jest.sh` zawarty w kolumnie **materiały** tabeli z materiałami na laboratorium,
- ★ polecenie `cowsay`

# Podsumowanie

## ● Zagadnienia podstawowe

1. Czy system X Window jest systemem bazującym na rozwiązaniach sieciowych?
2. Jakie są różnice pomiędzy tradycyjnym schematem programu graficznego, a modelem X Window?
3. W jaki sposób można uruchomić serwer X window?
4. W jaki sposób komunikują się ze sobą klient i serwer X?
5. Czy na jednym komputerze może działać jednocześnie więcej niż jeden serwer X?
6. W jaki sposób można kontrolować dostęp do serwera X?
7. Czy można uruchomić klienta X na innym komputerze niż ten, na którym działa serwer?
8. Czy termin zarządca okien jest jednoznaczny z serwerem X Window?
9. W jaki sposób przekazać klientowi adres serwera X?
10. W jaki sposób ustawić zmienną środowiskową DISPLAY?
11. Do czego służy zarządca okien? Podaj kilka przykładów.
12. Jak uruchomić klienta X Window na zdalnym komputerze?
13. Wymień kilka klientów X Window. Do czego służą?
14. Jakie mogą pojawić się argumenty wywołania klienta X Window?
15. Co oznacza zapis `xcalc -display 156.17.9.1:0`?
16. Jak wyświetlić listę procesów użytkownika?
17. Do czego służy plik `.xinitrc`?

18. Jakie są różne sposoby (5) uruchamiania skryptów powłoki użytkownika? Czym one się różnią?
19. Jakie wartości powinny być zwracane w funkcji `int main()` programu napisanego w języku C, by po jego skompilowaniu, poniższa instrukcja dawała prawidłowy rezultat:  

```
if ./a.out >/dev/null; then echo Działa poprawnie; else\  
    echo Wystąpił błąd; fi
```
20. W jaki sposób (podaj 3 różne) za pomocą polecenia `mkdir` można utworzyć katalog o nazwie zawierającej spację?
21. Jaka jest różnica pomiędzy wywołaniami  

```
prog1 ; prog2  
prog1 && prog2  
prog1 || prog2
```

## ● Zagadnienia rozszerzające

1. W jaki sposób można zatrzymać oraz zrestartować system X Window?
2. Jaką zaletę posiada zarządca okien `twm` w porównaniu do `gnome`?
3. Jakie jeszcze, poza podanymi na wykładzie, mogą być opcje wywołania klienta X Window?
4. Jacy zarządcy okien są udostępnieni w Twojej dystrybucji Linuksa? Czym się pomiędzy sobą różnią?
5. Jakie atrybuty, poza `rwx` można nadawać plikom? Do czego one służą?

6. W jaki sposób można pokazać zastosowanie lokalizacji? (zaproponuj zestaw poleceń, które zwrócą różne wyniki w zależności od ustawień zmiennych lokalizacyjnych)

## ● Zadania

1. Połącz się zdalnie z komputerem `diablo` z przekazaniem X serwera do ekranu drugiego. Uruchom na nim program `xclock`.
2. Spróbuj uruchomić klienta X Window na serwerze `diablo`.
3. Uruchom program `xterm` w lewym górnym rogu ekranu oraz `xclock` w prawym górnym rogu ekranu z zadaną geometrią (dobraną eksperymentalnie).
4. Wypróbuj parę możliwości wywołania klientów X Window z zastosowaniem opcji wywołania.
5. Zapisz jednolinijkowe polecenie które wykona kolejno wszystkie poniższe czynności:
  - ★ uruchomi program `p1` podając na jego wejście zawartość pliku `we.txt`,
  - ★ wyjście programu `p1` posortuje i przekaże na wejście programu `p2`,
  - ★ wynik `p2` dopisze do pliku `wy.txt`,
  - ★ jeśli powyższe czynności zakończą się sukcesem, uruchomi w tle edytor `emacs` dla pliku `wy.txt` oraz wypisze w konsoli komunikat o poprawnym zakończeniu.W testach jako program `p1` można wykorzystać polecenie `cat`, zaś jako program `p2` polecenie `uniq` (ewentualnie `wc`).