

Wprowadzenie

Wojciech Myszka

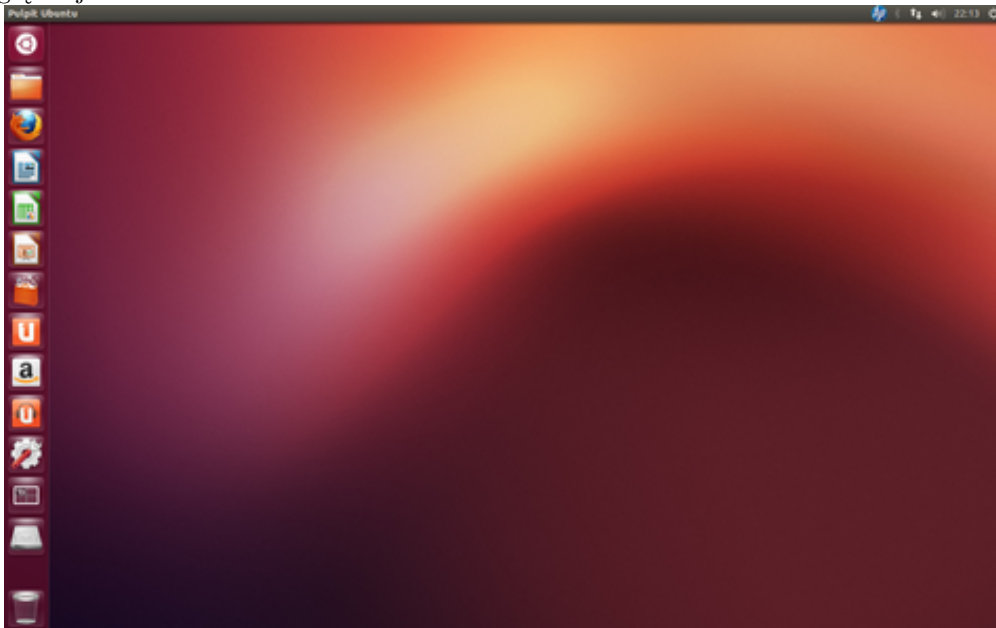
1 Środowisko pracy

Zajęcia odbywają się w salo 604/B1 (na poddaszu). Nie da się bezpośrednio tam tam dojechać windą!

Na komputerach w laboratorium zainstalowany jest system Ubuntu¹. Różni się ono nieco od środowiska Windows. W związku z tym pozwoliłem sobie poświęcić mu kilka zdań.

1.1 Logowanie do systemu

Aby zalogować się, należy na stronie wybrać „Zaloguj”, następnie wpisać swój numer indeksu i, następnie, hasło. Po zalogowaniu się (pierwsze logowanie może trwać nieco dłużej) ekran będzie wyglądał jakoś tak:





Jest to widok „pulpitu”. Po lewej stronie znajduje się „pasek uruchamiania” (*launcher*). Na pasku umieszczone są ikony standardowych aplikacji. Kliknięcie w ikonę uruchamia aplikację. **Uwaga: Klikamy tylko raz! Podwójne kliknięcie może powodować poważne kłopoty!**

W razie kłopotów można się zalogować jako „Gość”. Zwracam uwagę, że konto Gość jest kontem **wirtualnym**, po wyłączeniu komputera wszystkie pliki przepadają! Poza tym jest najzupełniej funkcjonalne, pliki można nagrywać na nośnikach zewnętrznych (podłączonych przez USB).

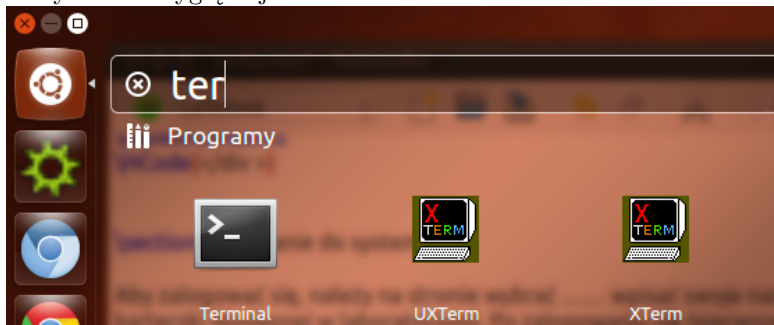
¹W chwili pisania tych słów jest to 12.04.5 LTS.

1.2 Uruchamianie aplikacji

Zwracam uwagę na ikonę  kliknięcie w nią pozwala wybierać aplikacje do uruchomienia. Trochę podobne działanie ma równoczesne naciśnięcie klawiszy „windows” i klawisza „A”. Jednak w tym drugim przypadku dostajemy dostęp do znacznie szerszego zestawu zainstalowanych aplikacji. Tak, czy inaczej — wyszukiwanie aplikacji po ikonce jest dosyć niewygodne. Najlepiej zacząć pisać jej nazwę. . .

Założmy, że chcemy uruchomić aplikację terminal (zaraz będzie nam potrzebna). Po wpisaniu kilku pierwszych liter mamy kilka aplikacji o nazwie z nimi zgodnej. Możemy wybrać jedną z nich za pomocą myszy lub pierwszą naciskając terminal. Dostęp do funkcji związanych z tą ikoną można uzyskać również po naciśnięciu klawisza „windows”². Zatem aby uruchomić terminal wystarczy nacisnąć klawisz  napisać „ter” i nacisnąć klawisz Enter. Albo jeszcze prościej naciskając równocześnie trzy klawisze Ctrl-Alt-T. . .

Wszystko to wygląda jakoś tak:



1.3 Zmiana hasła

Aby zmienić hasło należy wykonać następujące czynności:

1. Otworzyć terminal (jak to zrobić opisano powyżej).
2. W terminalu wydać polecenie `yppasswd` i odpowiedzieć na dwa pytania:

```
Changing NIS account information for xxxxx on xeon4.immt.pwr.wroc.pl.  
Please enter old password: Changing NIS password for xxxxx on xeon4.immt.pwr.wroc.pl.  
Please enter new password: Please retype new password:
```

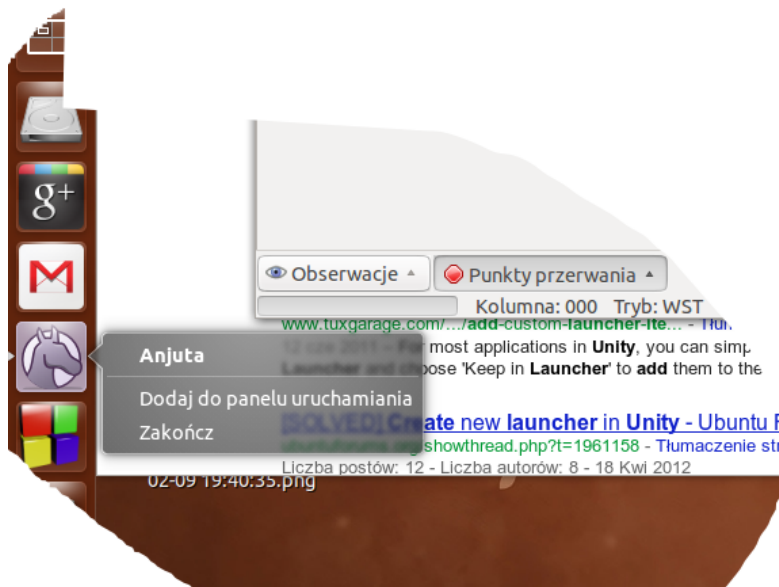
czyli najpierw podać stare hasło, a następnie **dwa razy** podać hasło nowe.

Uwaga: Wpisywanych informacji nie będzie widać!

1.4 Dodawanie/usuwanie aplikacji do/z paska uruchamiania

Aplikacje, z których korzystamy najczęściej można dodać do paska uruchamiania. Pierwszy raz aplikację uruchamiamy standardowo (patrz rozdział 1.2). Ikona aplikacji pojawi się na pasku uruchamiania. Klikamy w nią prawym klawiszem myszy i z menu wybieramy „Dodaj do panelu uruchamiania”:

²W terminologii linuxowej nazywa się on *Super*.



W podobny sposób możemy aplikację usunąć z paska (pozostanie w systemie i będzie można ją uruchamiać):



Z listy wybieramy „Usuń z panelu uruchamiania”.

2 Środowiska programistyczne

Jednym ze sporych ułatwień w programowaniu jest tak zwane zintegrowane środowisko programistyczne. Na szybko opiszę tu kilka takich programów zainstalowanych w pracowni 604 B1.

Idea wszystkich Zintegrowanych Środowisk Programistycznych (IDE — Integrated Development Environment) oparta jest na pojęciu projektu. Projekt to jeden lub więcej plików źródłowych wraz z zasadami budowy programu wynikowego. Każdy z plików źródłowych kompilowany jest niezależnie, a następnie pliki łączone są w gotowy program.

Po dokonaniu zmian, trzeba przekompilować jedynie te pliki, które uległy modyfikacji. Pozwala to w pewnych sytuacjach (gdy program jest bardzo duży) znacznie przyspieszyć cały proces

Wydaje się, że na potrzeby naszego laboratorium wystarczy środowisko najprostsze, czyli Geany w wersji „ad-hoc” (prawdę mówiąc projekty tam nie działają zupełnie. . .)

Opisywane środowiska to:

- Geany
- CodeLite
- Anjuta
- Eclipse
- Code::Blocks Jest to bardzo sympatyczne środowisko programistyczne (cieszące się dużym zainteresowaniem w niektórych środowiskach Politechniki Wrocławskiej). Nie korzystałem z niego, nie mam żadnego własnego opisu. Skorzystać można z dostępnej dokumentacji oraz zajrzeć na jego stronę domową.

Poniżej informacja jak piszą programy **prawdziwi programiści**.

Otóż prawdziwi programiści korzystają z terminala i jakiegoś tekstowego edytora; pod linuxem, podstawowym edytorem jest uruchamiany w terminalu **vim** (lub jego graficzny odpowiednik **gvim**). Wpisują program do pliku, którego nazwa kończy się rozszerzeniem .c (na przykład ala.c), a następnie kompilują go poleceniem (wydanym w terminalu)

```
gcc -o ala ala.c
```

a później uruchamiają poleceniem

```
./ala
```

3 Schematy blokowe

Czasami mamy potrzebę (albo konieczność) stworzenia schematu blokowego. Jest bardzo wiele programów ułatwiających tę pracę. W laboratorium 604/B1 zainstalowany jest program dia. Program dostępny jest dla linuxa, Mac Os i Windows.

Jak wiadomo schemat blokowy (flowchart, block diagram) to zestaw geometrycznych bloków połączony liniami służący do zobrazowania (w naszym przypadku) algorytmu. I w tym celu możemy/będziemy go wykorzystywać.

Krótki opis programu dia.

4 Upiększanie kodu

Bardzo często, w ferworze walki, nie zastanawiamy się jak wygląda nasz program. Chwile później okazuje się, że strasznie trudno zrozumieć o co w nim chodzi. Czasami pomaga upiększenie go. Czyli takie przetworzenie, żeby wyglądał ładniej i był bardziej czytelny.

Oto przykład:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int m=144;
    int n=12;
    int r;
    while (r!=0)
    {
        r=m/n;
        m=n;
        n=r;
    }
    printf ("Największy wspólny dzielnik , to : %d\n",m);
    getchar();
    return 0;
}
```

i po przetworzeniu:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int m = 144;
    int n = 12;
    int r;
    while (r != 0)
    {
        r = m % n;
        m = n;
        n = r;
    }
    printf ("Największy wspólny dzielnik , to : %d\n",m);
    getchar();
    return 0;
}
```

Różnica nie jest wielka, ale czasami warto popatrzeć na program innym okiem.

Ja do „upiększania kodu” używam programu uncrustify wspomagając się graficznym interfejsem UniversalIndentGUI. Programy darmowe, dostępne zarówno w środowisku Linux jak i

Windows...

5 Dostęp do serwera

Wszystkie pliki znajdują się na serwerze. Jego adres to **xeon4.immt.pwr.wroc.pl**. Aby przesyłać plik należy użyć programu WinSCP. Program w wersji „Portable executables” nie wymaga instalacji. W przypadku systemów linuxowych wystarczy program scp. Aby uzyskać „tekstowy” dostęp do serwera należy użyć programu putty. Sugeruję pobierać wersję „exe”. W przypadku systemów linuxowych używamy poleceń ssh/slogin.

6 Wersja PDF tego dokumentu...

...pod adresem.

Wersja: 40 **z drobnymi modyfikacjami!** data ostatniej modyfikacji 2016-04-22 09:42:18 +0200