

Klasa: II TI Technikum Kształtowania Środowiska - Technik Informatyk

SYSTEMY OPERACYJNE

Temat: Edytor tekstu VI.

Wykonałam zrzuty z Podręcznika: WSiP „Przygotowanie stanowiska komputerowego. Część2” T.Marciniuk, K.Pytel, S.Osetek. Proszę zapoznać się z treścią tego podręcznika.

Przeczytaj rozdział dotyczący dzisiejszej lekcji.

160

PRZYGOTOWANIE STANOWISKA KOMPUTEROWEGO DO PRACY. CZĘŚĆ 2

18

Programy narzędziowe w środowisku tekstowym systemu Linux

ZAGADNIENIA

- Jak posługiwać się edytorem tekstu vi?
- Jak tworzyć skrypty powłoki?
- Na czym polega kompresja i archiwizacja?
- W jaki sposób tworzy się archiwa w systemie Linux?
- W jaki sposób skompresować i zdekompresować pliki i katalogi?
- Gdzie znaleźć publikacje elektroniczne dotyczące systemu Linux?

18.1. Edytor tekstu vi

Edytor vi jest dostępny we wszystkich klonach systemu Unix / Linux, a w większości systemów jest edytorem domyślnym. Niektóre programy, np. **cron**, wykorzystują vi, jako swój bazowy edytor opcji. Sposób obsługi vi jest różny od sposobu obsługi edytorów pracujących w środowisku Windows. Każdy użytkownik Linuksa powinien więc poznać vi przynajmniej w takim stopniu, aby mógł wykorzystać go do edycji plików konfiguracyjnych. Istnieje w zasadzie tylko jedna metoda nauczenia się obsługi – trzeba ćwiczyć w praktyce. Na potrzeby początkujących użytkowników vi opracowano samouczek, który w ciągu kilkadziesiąt minut pozwoli na przećwiczenie podstawowych możliwości edytora. Interaktywny samouczek programu vi w wersji angielskiej jest dostępny między innymi na stronie <http://www.openvim.com/tutorial.html> lub w wersji polskiej na stronie <http://jaki-linux.org/aplikacje/uzywaj-vim-a-jak-profesjonalista-cz-1/>. Aby poznać najczęściej używane, należy uruchomić samouczek i wykonać wskazane ćwiczenia.

Istnieje kilka wersji edytora vi. Najpopularniejsza i najbardziej rozbudowana jest wersja VIM, która ma też najwięcej opcji i możliwości, lecz zasady pracy w obu programach są podobne. Edytor vi pracuje w środowisku tekstowym; może być uruchomiony poleceniem **vi**. Jeżeli trzeba utworzyć nowy lub otworzyć istniejący plik, można od razu podać jego nazwę, np. **vi plik.txt**.

Na ekranie można wydzielić dwa obszary robocze: **obszar poleceń** (ostatnia linia na dole ekranu) i **obszar edycji tekstu** (pozostałe linie). Edytor vi ma dwa tryby pracy: tryb poleceń do wydawania poleceń określających, co chce się zrobić, oraz tryb edycji, w którym edytuje się tekst. Po uruchomieniu vi domyślnie znajduje się w trybie poleceń. Aby przejść do trybu edycji, należy nacisnąć klawisz [INS] (insert); aby przejść do trybu poleceń, należy nacisnąć klawisz [Esc].

Aby zakończyć pracę z vi, należy nacisnąć: [Esc] :q! – bez zapisu tekstu do pliku, [Esc] :wq – z zapisem tekstu do pliku.

Przemieszczanie się po tekście odbywa się za pomocą klawiszy kursorów w obu trybach pracy, a dodatkowo w trybie poleceń można użyć przycisków [h] – lewo, [j] – dół, [k] – góra, [l] – prawo. Liczba opcji, jakimi dysponuje vi, jest bardzo duża. Aby poznać najczęściej używane, należy uruchomić samouczek i wykonać wskazane ćwiczenia.

18.2. Skrypty powłoki

Skrypty są zwykłymi plikami tekstowymi, w których są zapisane polecenia zrozumiałe dla powłoki. Zadaniem powłoki jest przetłumaczenie ich na polecenia systemu. Aby przygotować skrypt, należy:

- stworzyć plik, w którym trzeba umieścić kod, np.:

```
touch naszskrypt.bat
```

- za pomocą dowolnego edytora tekstu wpisać do pliku kolejno wykonywane polecenia;
- nadać uprawnienia do wykonywania pliku dla uprawnionych użytkowników;
- uruchomić skrypt:

```
./naszskrypt.bat
```

gdzie symbol ./ oznacza, że skrypt znajduje się w bieżącym katalogu.

Przykładowy, bardzo prosty skrypt może się składać z następujących poleceń:

```
#!/bin/bash
#Tu jest komentarz.
echo „Hello World”
```

Pierwsza linia skryptu zaczynająca się od znaków #! ma szczególne znaczenie – wskazuje na rodzaj powłoki, w której skrypt ma być wykonany. Tutaj skrypt zawsze będzie wykonywany przez interpreter poleceń **/bin/bash**, niezależnie od tego, jakiego rodzaju powłoki używamy w danej chwili.

Znak # (hasz) oznacza komentarz – wszystko, co znajduje się za nim w tej samej linii, jest pomijane przez interpreter.

Trzecia linia spowoduje wydrukowanie na standardowym wyjściu (*stdout*), czyli na ekranie, napisu: „Hello World”.

Między nawiasami a treścią warunku muszą być spacje, tak jak powyżej.

Polecenie **test** zwraca wartość **0** (*true*), jeśli warunek jest spełniony, i wartość **1** (*false*), jeśli warunek nie jest spełniony, np. **test -e plik**

Wybrane przykłady operatorów polecenia **test**:

- e** plik istnieje;
- =** sprawdza, czy wyrażenia są równe;
- !=** sprawdza, czy wyrażenia są różne;
- d** wyrażenie istnieje i jest katalogiem;
- r** można czytać plik;
- w** można zapisywać do pliku;
- x** można wykonać plik;
- lt** mniejsze niż;
- gt** większe niż;
- ge** większe lub równe;
- le** mniejsze lub równe.

instrukcja **case** – pozwala na dokonanie wyboru spośród kilku wzorców.

Sprawdzana jest wartość zmiennej po słowie kluczowym **case** i porównywana ze wszystkimi wariantami po kolei. Jeśli dopasowanie zakończy się sukcesem, wykonane zostanie polecenie lub polecenia przypisane do danego wzorca. W przeciwnym wypadku zostanie użyte polecenie domyślne oznaczone gwiazdką „***”**. Składnia polecenia:

```
case zmienna in
„wzorzec1”) polecenie1 ;;
„wzorzec2”) polecenie2 ;;
„wzorzec3”) polecenie3 ;;
*) polecenie_domyślne
esac
```

PRZYKŁAD 18.1

```
#!/bin/bash
echo „Podaj cyfrę dnia tygodnia”
read d
case „$d” in
„1”) echo „Poniedziałek” ;;
„2”) echo „Wtorek” ;;
„3”) echo „Środa” ;;
„4”) echo „Czwartek” ;;
„5”) echo „Piątek” ;;
„6”) echo „Sobota” ;;
„7”) echo „Niedziela” ;;
*) echo „Nic nie wybrałeś”
esac
```

- pętla **for** – wykonuje polecenia zawarte wewnątrz pętli, na każdym składniku listy. Składnia polecenia:

```
for zmienna in lista
do
polecenie
done
```

Praca domowa. Opisz twórcę systemu Linux i przedstaw wszystkie dystrybucje Ubuntu od najstarszej do najnowszej w pliku word, do dnia 15.05.2020r. (piątek) na adres **bodzia.koc@vp.pl**

Pozdrawiam 2TI

Bogusława Kocątek