

Systemy Operacyjne

Bartłomiej Koczenasz

KATALOG TO NIE FOLDER!!!

Katalog jest nieodłącznie związany z systemem plików i jest on fizycznie przechowywany na dysku. **Folder** natomiast może być wirtualny i istnieć tylko w obrębie danego systemu operacyjnego. Przykładami folderów w systemie Windows są okno Mój komputer czy Panel sterowania, które jednak nie istnieją na dysku, lecz jedynie w systemowym rejestrze. Ponadto jeden plik może być przechowywany tylko w jednym katalogu, ale już w kilku różnych folderach jednocześnie.

Czym jest system operacyjny?

Systemem operacyjnym nazywamy zorganizowany zbiór programów pośredniczących między sprzętem komputerowym a jego użytkownikiem, który dokonuje dynamicznego rozdziału dostępnych zasobów pomiędzy istniejące procesy w sposób zapewniający ich poprawne oraz efektywne, w sensie przyjętego kryterium, wykorzystanie.

System komputerowy to sprzęt komputerowy oraz system operacyjny + dodatkowe oprogramowanie, traktowane jako całość służąca do ściśle określonych celów.

Zasobem w systemie operacyjnym nazywamy każdy rodzaj sprzętu komputerowego lub danych, niezbędny do prawidłowego rozpoczęcia, wykonywania oraz zakończenia istniejących procesów.

Zasób jest **nieodnawialny**, jeśli jest zużywany przez wykorzystujący go proces.

Zasób jest **nieprzywłaszczalny**, jeśli nie jest możliwe zawieszenie jego wykorzystywania przez aktywny proces.

Zasób jest **niepodzielny**, jeśli wykorzystywać go może w danej chwili tylko jeden proces

Funkcje systemu operacyjnego

Trzy główne funkcje OS to:

- 1) Sterowanie procesami
- 2) Zarządzanie pamięcią
- 3) Zarządzanie zasobami

Maszyna wirtualna

Maszyna wirtualna to abstrakcyjny model komputera o innych właściwościach aniżeli maszyna fizyczna; składają się na nią rozwiązania dotyczące obsługi urządzeń wejścia/wyjścia, pamięci operacyjnej (wirtualnej), systemu plików, sterowania programami (procesami) oraz przyjęte metody ochrony danych

Struktury systemów operacyjnych

Struktura monolityczna (jednolita) - system operacyjny jest zbiorem usług, z których każda może wywołać inną

Struktura warstwowa- system operacyjny jest podzielony na warstwy odpowiadające za poszczególne funkcje systemu. Główne warstwy systemu o budowie warstwowej to warstwa **jądra**, warstwa **powłoki**, warstwa **programów użytkownika**

Tryby pracy systemu operacyjnego

SO może pracować w różnych trybach, które różnią się między sobą zbiorem usług dostępnych dla użytkownika.

Tryb pracy SO decyduje o tym, jakie usługi SO dany proces może wywołać.

Wyróżniamy dwa główne tryby: użytkownika i jądra

Klasyfikacje systemów operacyjnych

Systemy operacyjne: jednoprogramowe, wieloprogramowe

Systemy operacyjne: dla jednego użytkownika, dla wielu użytkowników

Sieciowe systemy operacyjne

Rozproszone systemy operacyjne

Systemy operacyjne czasu rzeczywistego

Systemy operacyjne ogólnego przeznaczenia: wsadowe, wielodostępne

Systemy operacyjne do obsługi baz danych

Cechy współczesnych systemów operacyjnych

Architektura oparta o mikrojądro

Stosowanie techniki „klient-serwer”

Wielowątkowość

Wieloprzetwarzanie

Obliczenia rozproszone

Pojęcie procesu

Procesem nazywamy obiekt w systemie operacyjnym

- któremu przydzielany jest procesor,
- który zawiera kod programu wykonywanego w trakcie swojego istnienia,
- posiada własny obszar pamięci operacyjnej,
- dysponuje zbiorem niezbędnych zasobów.

Procesy

Wyróżniamy procesy (zadania):

sekwencyjne (wykonywane na 1 procesorze)

równoległe (wykonywane na wielu procesorach)

Z wykonywaniem wielu procesów jednocześnie wiążą się dwa pojęcia:

wieloprogramowość (ang. multiprogramming) - naprzemienne wykonywanie „fragmentów” różnych procesów przez 1 procesor

wieloprzetwarzanie (ang. multiprocessing) - współbieżne wykonywanie różnych procesów na różnych procesorach

SO operacyjny tworząc nowy proces tworzy jego obraz. Informacje dotyczące procesów można podzielić na:

lokalne - obsługiwane przez PCB (Process Control Block),

globalne - obsługiwane przez tabelę procesów.

Procesy

Proces macierzysty - proces tworzący (za pomocą funkcji systemowej) nowy proces, zwany jego procesem .

Współbieżne procesy macierzyste i potomne mogą być:

asynchroniczne – wykonują się niezależnie od siebie,

synchroniczne – ich wykonania zależą od siebie

Proces – sierota (ang. orphan) - proces potomny istniejący w chwili, gdy nie istnieje już jego proces macierzysty (rzadko się zdarza, w niektórych SO w ogóle, np. Unix)

Proces - upiór (ang. zombie) - proces od chwili jego zakończenia (zabicia) do chwili powiadomienia jego procesu macierzystego o tym fakcie (w niektórych SO, jest to normalny stan, np. Unix)

Zagłodzenie procesu – nie wykonywanie procesu gotowego, mimo iż zdarzenie na które on czeka, wystąpiło dowolną liczbę razy.

Demon - proces biegnący w tle, wykonujący okresowo określone czynności (np. sprawdzanie skrzynki pocztowej, uruchamianie skryptów w określonym czasie itp.)

Sekcja krytyczna - fragment kodu procesu, poprzez który uzyskuje on dostęp do niepodzielnego, nieprzywłaszczalnego zasobu.

System plików

Plik to systemowa struktura danych, za pomocą której SO dokonuje operacji na fizycznym zbiorze danych

Fizyczny zbiór danych to dowolny zbiór danych skończonej wielkości, posiadający unikatową nazwę oraz zapisany zazwyczaj w pamięci zewnętrznej

System plików

System plików to część systemu zarządzania pamięcią, odpowiedzialna za operacje wykonywane na plikach, ich przechowywanie oraz kontrolę współdzielenia plików.

Katalog to plik zawierający informacje o innych plikach, zwykle specjalnie chroniony przez system.

Główne techniki obsługi urządzeń I/O

- **praca pośrednia** (ang. off-line) – polega na wykorzystaniu jednostek taśmy magnetycznej jako urządzeń I/O, zamiast czytników kart oraz drukarek (dzisiaj praktycznie niespotykana), praca bezpośrednia: czytnik kart → CPU → drukarka, praca pośrednia: czytnik kart → taśma → CPU → taśma → drukarka,
 - **buforowanie; bufor** - ciągły obszar pamięci operacyjnej o wielkości możliwiającej przechowanie jednego lub kilku rekordów pliku jednocześnie
- Buforowanie** – technika obsługi urządzeń I/O, polega na jednoczesnym wykorzystywaniu przez aktywny proces zarówno CPU, jak i urządzenia I/O
- Spooling** – technika obsługi urządzeń I/O, polegająca na wykorzystaniu dysku jako bufora

Metody obsługi I/O

Odpytywanie – metoda obsługi urządzeń I/O, polegająca na cyklicznym sprawdzaniu przez CPU stanu wszystkich urządzeń I/O

Przerwanie – sygnał będący reakcją na zdarzenie, powodujące zmianę trybu pracy procesora (dzielimy na: sprzętowe i programowe)

DMA – technika obsługi urządzeń I/O, polegająca na przesyłaniu danych z (do) urządzenia I/O bezpośrednio do (z) PAO, z pominięciem CPU

Podstawowe pojęcia ochrony danych

- **ochrona danych** – zapewnienie bezpieczeństwa danych na poziomie różnych warstw SO
- **bezpieczeństwo** – zapewnienie danym ochrony przed przypadkowym lub celowym uszkodzeniem, odkryciem lub modyfikacją
- **tajność** – prawo do określenia informacji jako zakrytej (tzn. niedostępnej dla innych)
- **poufność** – status danych, określający stopień ich ochrony

Klasyfikacja Flynna

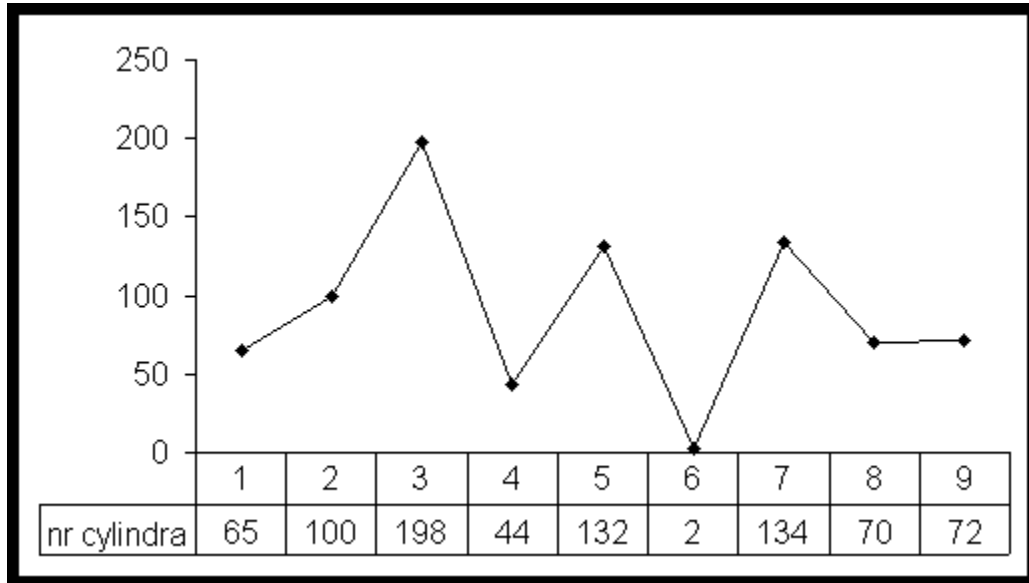
- **SISD** (ang. Single Instruction, Single Data) - pojedyncza jednostka przetwarzająca (CPU) wykonuje pojedynczy ciąg rozkazów.
- **SIMD** (ang. Single Instruction Stream, Multiple Data Stream) - kilka synchronicznie działających CPU wykonuje ten sam ciąg rozkazów na różnych strumieniach danych
- **MIMD** (ang. Multiple Instruction, Multiple Data) – kilka asynchronicznie działających CPU wykonuje różne ciągi rozkazów na różnych strumieniach danych
- **MISD** (ang. Multiple Instruction, Single Data) - wiele strumieni rozkazów wykorzystuje ten sam strumień danych. Model MISD nie jest wykorzystywany w praktyce ze względu na brak zastosowań

Algorytmy

Metody planowania dostępu do dysku

Metoda FCFS

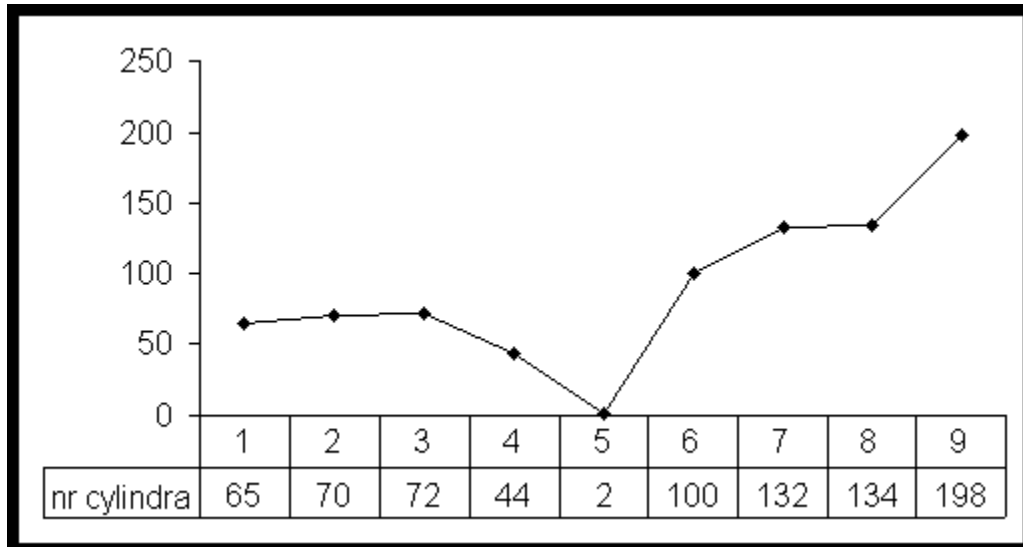
Jest to najprostszy algorytm planowania dostępu do dysku. Zamówienia są wykonywane w kolejności ich zgłaszania. Skrót FCFS (first-come, first-served) oznacza 'pierwszy zgłoszony - pierwszy obsłużony'. Jest to najbardziej sprawiedliwy z algorytmów jednak nie najszybszy. Załóżmy że kolejka zamówień na operacje we-wy wygląda tak i odnosi się do bloków w cylindrach: 100, 198, 44, 132, 2, 134, 70, 72



Metody planowania dostępu do dysku

Metoda SSTF

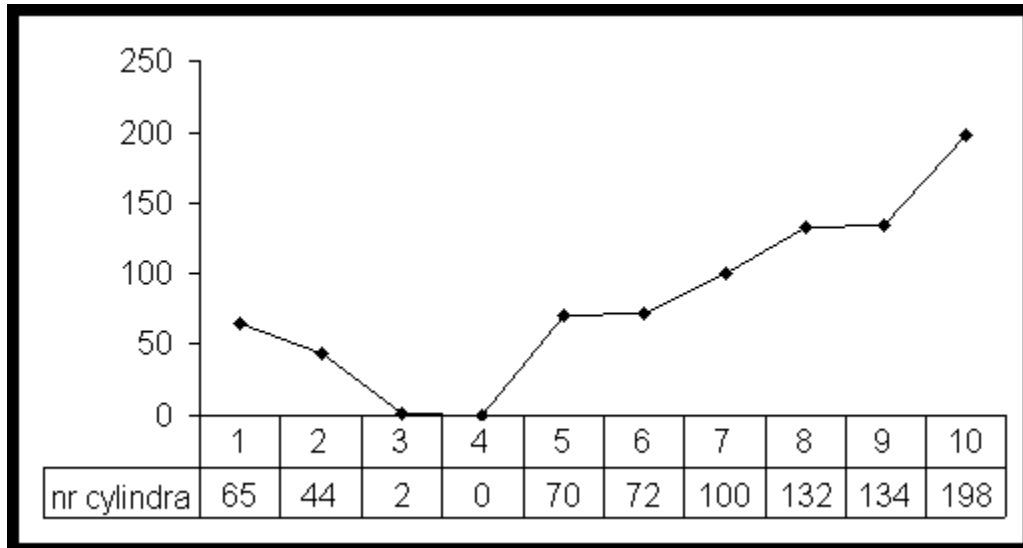
Skrót SSTF jest skrótem od Shortest-Seek-Time-First, co oznacza -najpierw najkrótszy czas przeszukiwania. Metoda ta polega na realizacji zamówień w takiej kolejności, aby wielkość liczby cylindrów pomiędzy każdym zamówieniem była jak najmniejsza. Startując od cylindra 65 mamy:



Metody planowania dostępu do dysku

Metoda SCAN

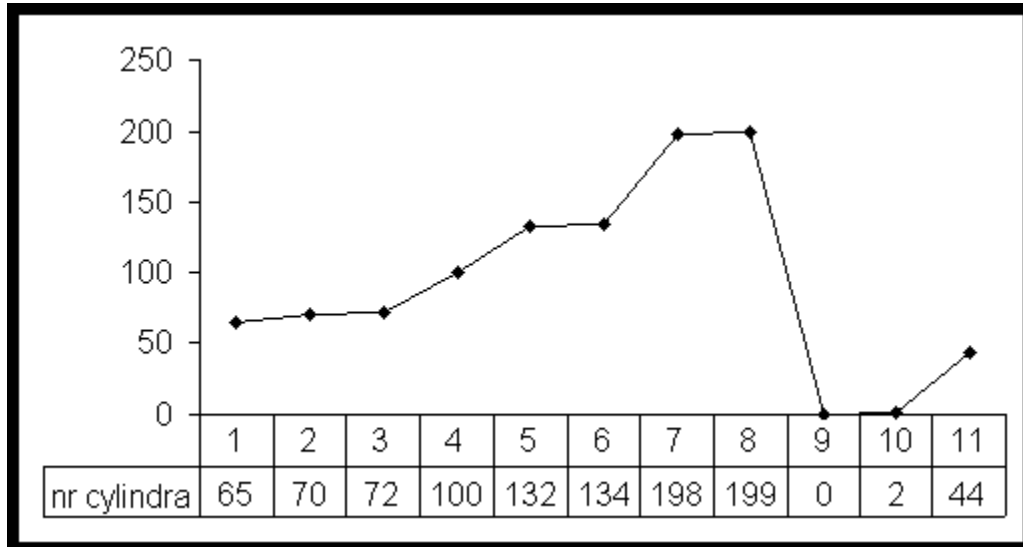
Algorytm SCAN polega na tym, że ramię dysku rozpoczyna przemieszczanie się od jednej krawędzi dysku, do przeciwległej krawędzi. W czasie przemieszczania się ramienia dysku, obsługiwane są zamówienia w kolejności napotkanych, żądanych przez system, cylindrów. Po osiągnięciu przeciwległej krawędzi ramię dysku zmienia kierunek ruchu i zaczyna wracać, obsługując następne zamówienia napotkane po drodze.



Metody planowania dostępu do dysku

Metoda C-SCAN

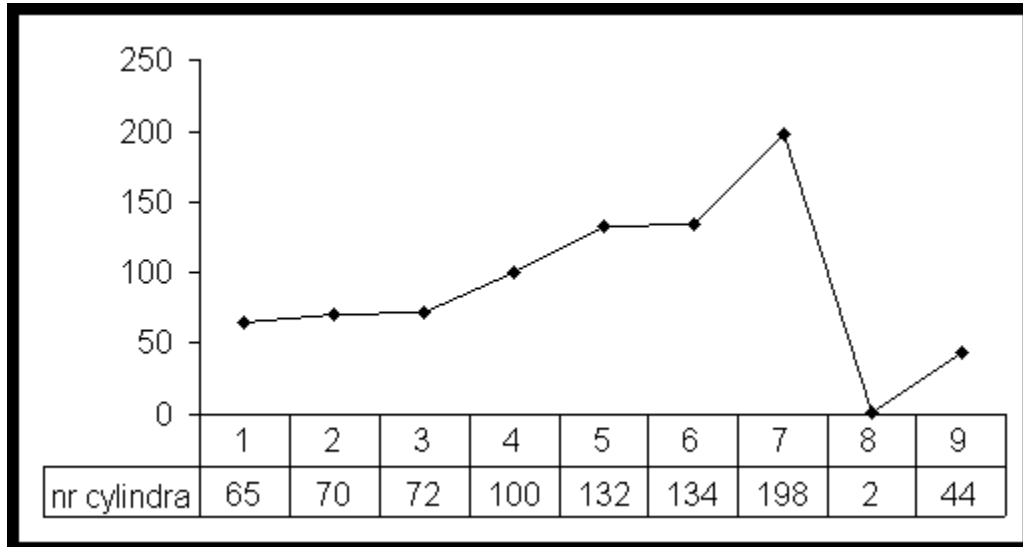
Odmianą algorytmu SCAN, jest algorytm C-SCAN (cilcural SCAN). Polega ona na tym, że po osiągnięciu jednej krawędzi dysku głowica wraca do drugiej krawędzi, ale nie wykonuje nic poza tym. Po osiągnięciu tej drugiej krawędzi, głowica przemieszcza się do pierwszej krawędzi, wykonując zlecenia po drodze. Algorytm ten jest zaprojektowany w trosce o bardziej równomierny rozkład czasu czekania na wykonanie zlecenia.



Metody planowania dostępu do dysku

Metoda LOOK

Zarówno w planowaniu SCAN jak i C-SCAN głowica przemieszcza się od jednego skrajnego położenia do drugiego skrajnego. Żaden z tych algorytmów jednak w rzeczywistości nie jest tak implementowany. Przeważnie głowica przesuwa się pomiędzy skrajnymi zamówieniami. Gdy osiągnie jedno z nich, natychmiast robi zwrot i idzie w kierunku przeciwnym, nie dochodząc do skrajnego położenia na dysku. Są to algorytmy LOOK i C-LOOK które są poprawionymi odpowiednikami algorytmów SCAN i C-SCAN.



Algorytmy przydziału procesora

EDD:

Każde zadanie ma jakąś linię krytyczną (czas przed którym musi zostać wykonany). Sortujemy je od tego który ma najmniejszą wartość tej linii do tego o największej. Dzięki temu jest największa szansa że większość z nich zdąży się wykonać przed wyznaczonym czasem.

ERT:

Porządkujemy zadania niemalejąco względem czasu gotowości.

LS:

Tutaj po prostu przydzielamy je tak jak są podane do pierwszego wolnego procesora.

LPT:

Najpierw układamy zadania od największego czasu wykonywania do najmniejszego, a potem stosujemy LS.

Round-robin:

Procesor wykonuje przez chwilę część procesu (np. jakiś wątek) przez określony czas, a następnie przełącza się na kolejny na liście. Po dojechaniu do końca listy zaczyna od początku.

SPT:

Algorytm który układa zadania względem czasu wykonywania, czyli najpierw jest ten który najszybciej się zakończy, aż do tego który będzie działał najdłużej.

Komendy

Komendy UNIX

ssh login@domena/ip- logowanie do konta ssh

cat a.txt- wyświetla zawartość pliku a.txt na ekranie

chmod a-rwx lub **u-r, g-w, o+x** lub **763 a.txt**- ustawia prawa dostępu

cp a.txt b.txt- kopiuje plik

ln a b- tworzy dowiązanie b z pliku a. opcje: **-s**(symboliczne - katalogi)

mv a.txt b.txt- przenosi plik/zmienia nazwę

rm a.txt- usuwa plik. opcje: **-f**(nie pyta się czy na pewno) **-r** (usuwa katalog z podkatalogami)

umask 000- ustawia domyślne ustawienia praw dostępu do plików

cd katalog1 - przejście do podanej ścieżki

ls- pokazuje pliki w katalogu opcje: **-a** wyświetla wszystkie pliki **-l** listuje pliki

mkdir katalog1- utworzenie katalogu

pwd- sprawdza w jakim jesteśmy katalogu

rmdir katalog1- usunięcie katalogu

kill- „zabijamy” procesy

ps- pokazuje nam aktualnie wykonywane procesy

time- pokaze czas i zuzycie procesora

Komendy UNIX

tee- przekierowuje zarówno na ekran jak i do pliku

man polecenie- instrukcja polecenia

--help- pomoc

cmp a.txt b.txt- porównuje dwa pliki

diff- porównuje pliki i wypisuje różnice na standardowym wyjściu

grep 'Ala' plik.txt- wyszukuje w pliku linie w których występuje Ala. opcje: **-v** neguje

grep 'A[lg]a' plik.txt- Aga lub Ala

grep 'A.a' plik.txt- Aga, Ala,Aaa,Aba itd.

grep '^Ala' plik.txt- posiadające Ala na początku

grep 'Go+gle'- gogle, google, goooogle itd.

grep '[0-9]+'- dowolna liczba

grep 'the' * >plik.txt- szuka w dowolnym dokumencie i przekazuje do pliku

head a.txt- wypisuje pierwsze 10 linii a.txt

more- wyświetla plik z przewijaniem w dół można stosować np. z catem jako | more

sort- sortuje linie w tekście. opcje: **-d** malejaco **-a** rosnąco(domyslne)

tail a.txt- wypisuje 10 ostatnich linii a.txt

Komendy UNIX

wc– zlicza znaki, słowa i linie opcje: **-c** znaki **-b** bity **-l** linie **-w** słowa.

bc– kalkulator binarny

cal- kalendarz

clear– czyści ekran

date– wyświetla datę

finger– szczegółowe informacje o użytkownikach

finger username- szczegółowe informacje o użytkowniku

passwd- zmiana hasła

who– sprawdzamy kto jest aktualnie zalogowany

whereis– szukanie gdzie jest plik, man, itp.

HOME- zmienna wskazująca na katalog domowy

PATH- zmienna określa ścieżkę przeszukiwania poleceń

PS1- określa ciąg znaków zachęty podstawowy

PS2- określa ciąg znaków zachęty dodatkowy

echo wiadomosc lub \$wiad– wypisuje na ekran 'wiadomosc' lub wartosc zmiennej wiad

env- uruchamia polecenie w zmodyfikowanej powłoczce

Komendy UNIX

...- link do nadkatalogu

#- przerywa ciąg znaków

" "- interpretuje znaki specjalne

' '- nie interpretuje znaków specjalnych

` ` - wykonuje polecenia w środku

shift- zmniejsza parametr o jeden

test- sprawdza czy zdanie logiczne jest prawdziwe i zwraca true lub false

eval- wykonuje polecenia

expr- wykonuje operacje matematyczne

read- oczekuje aż użytkownik wprowadzi wartość

touch a.txt- tworzy plik

write użytkownik- można napisać do kogoś

Komendy UNIX

gcc kod.c – kompiluje w konsoli kod w języku C

gcc -o program kod.c – kompiluje kod i tworzy plik o nazwie „program”

execl – zastępuje wykonywany aktualnie program na nowy program

exit – zakończenie procesu

fork – tworzy proces potomny

getpid – pobiera identyfikator PID procesu

getppid – pobiera identyfikator PID przodka procesu

wait – zawiesza przodka zamykanego procesu do zakończenia zamykania

fconfig – wyświetla status aktywnych interfejsów sieciowych

netstat – wyświetla info o sieci tj. TCP, porty, statystyki Ethernet, IPv4 IPv6

ping – diagnozuje połączenie sieciowe, wyświetla liczbę zgubionych pakietów oraz opóźnienia w transmisji

Komendy UNIX

export zmienna=1 – przekazuje do podpowloki zmienna

&- wykonanie polecenia w tle

&&- AND (2 polecenie jest wykonywane gdy pierwsze jest wykonane prawidłowo)

*****- dowolny zestaw znaków

?- dokładnie jeden dowolny znak

()- łączą komendy w grupy

{}- np. a{b,c,d} wynik: ab, ac, ad

[] - patrz. grep

| - przekierowanie wyjścia na wejście innego polecenia

||- OR

>- zapisuje do pliku(nadpisanie)

<- przekazanie wejścia polecenia do pliku

>>- zapisuje do pliku(dopisanie)

<<- przekazanie wejścia polecenia do czasu odczytania linii zawierającej zadany ciąg znaków

\$- odwołanie do zmiennej

**** - poprzedza znaki specjalne

Komendy MS-DoS

CMD – okno terminala

Start – uruchamia nowe okno terminala

Title – pozwala zmienić nazwę okna terminala

Attrib - wyświetla lub zmienia atrybuty pliku. Opcje a – plik archiwalny, r – plik tylko do odczytu, s – plik systemowy, h – plik ukryty

Copy – kopiuje plik

DEL/ERASE – usuwa plik

REN/RENAME- zmienia nazwę pliku/katalogu

XCOPY - kopiuje jeden lub wiele plików lub katalogów z podkatalogami

CD/CHDIR – zmiana katalogu

DIR – wyświetla listę katalogów

MD/MKDIR – tworzy katalog

RD/RMDIT – usuwa katalog

TREE – przedstawia graficzną strukturę katalogów na dysku lub w określonej ścieżce

Komendy MS-DoS

FC – porównuje 2 pliki i wyświetla różnicę między nimi

FIND – wyszukuje ciąg znaków w pliku

MORE – wyświetla informacje ekran po ekranie

SORT – sortuje wprowadzone dane i zapisuje wyniki

TYPE – wyświetla zawartość plików tekstowych

CLS – czyści okno

DATE – wyświetla datę i prosi o ew. Ustawienie nowej

EXIT – zamyka okno

PATH – wyświetla lub ustawia ścieżkę dla plików wykonywalnych

PROMPT – pozwala na utworzenie nowego komunikatu gotowości systemu

TIME – wyświetla czas i prosi o ew. Ustawienie nowego

IPCONFIG – wyświetla konfigurację interfejsów sieciowych

Komendy MS-DoS

SET – wprowadza zmienną

% - odwołanie do wartości zmiennej

/ - paramet poleceń

* - ciąg niepustych znaków

? – jeden niepusty znak

| - potok

> - przekazanie wyjścia

< - przekazanie wejścia

>> - przekierowanie strumienia I nadpisanie

\ - oddzielenie ścieżki

. – bieżący katalog

.. – katalog nadrzędny

Komendy MS-DoS

BREAK – włącza lub wyłącza możliwość przerywania pliku wsadowego przez ctrl+c

CALL – wywołanie z poziomu pliku wsadowego innego pliku wsadowego z ew. argumentem

ECHO – wypisanie komunikatu, wartości itp

FOR – pętla sterowana zmienną

GOTO – skok do miejsca oznaczonego podanego etykietą

IF – komenda warunkowa

PAUSE – Przerywa wykonywanie programu wsadowego do momentu naciśnięcia dowolnego klawisza.

REM – komentarz

SHIFT – przesunięcie argumentów pozycyjnych o 1 w lewo