

# **1. Wprowadzenie do systemów operacyjnych**

## **Wstęp**

Każdy system operacyjny ma do spełnienia dwa podstawowe cele: efektywne zarządzanie zasobami systemu komputerowego i zapewnienie wygodnej pracy użytkownikom. Nowoczesne systemy operacyjne są w większości systemami wielodostępnymi, które pozwalają pracować jednocześnie wielu użytkownikom za pośrednictwem terminali podłączonych bezpośrednio do komputera lub poprzez sieć komputerową.

Wykład 1 stanowi wprowadzenie do problematyki, którą przedstawiamy w niniejszym podręczniku. Zamieszczamy tu ogólną charakterystykę systemów operacyjnych ze szczególnym uwzględnieniem systemów wielodostępnych.

## 1.1. Ogólna charakterystyka systemów operacyjnych

Podstawowe informacje dotyczące systemów operacyjnych zostały zamieszczone w podręczniku Architektura systemów komputerowych [1]. Niniejszy podręcznik poświęcamy omówieniu w sposób bardziej szczegółowy złożonych zagadnień występujących w systemach wielodostępnych i wielozadaniowych. Zagadnienia te przedstawimy posługując się przykładem systemu Linux.

### Czym jest system operacyjny?

Nie istnieje ścisła definicja pojęcia systemu operacyjnego.

**Systemem operacyjnym** określane jest program lub zestaw programów, które pośredniczą między użytkownikami i ich programami a sprzętem komputerowym. System operacyjny to program sterujący wykonywaniem innych programów. Jest również administratorem zasobów systemu komputerowego.

System operacyjny składa się zwykle z dwóch części:

- jądra systemu,
- zestawu programów systemowych.

**Jądro** zawiera podstawowy kod systemu operacyjnego, niezbędny do jego działania. Jądro (lub jego rdzeń) jest przechowywane stale w pamięci operacyjnej komputera. **Programy systemowe** świadczą różne usługi systemowe i ułatwiają pracę użytkownikom.

### Cele systemu operacyjnego

Każdy system operacyjny ma do spełnienia trzy podstawowe cele:

- wykonywanie programów użytkowników,
- zapewnienie wygodnej pracy użytkownikom,
- efektywne zarządzanie zasobami systemu komputerowego.

Cele te pozostają niestety w sprzeczności ze sobą. Im bardziej "przyjazny" staje się system operacyjny, oferując użytkownikom wciąż więcej udogodnień, tym więcej zasobów sam zużywa, zamiast udostępniać je użytkownikom. Dotyczy to w szczególności zasobów pamięci operacyjnej i dyskowej oraz wykorzystania procesora. Osiągnięcie właściwego kompromisu jest szczególnie ważne w systemach wielodostępnych, gdy wielu użytkowników korzysta z ograniczonych zasobów systemu.

### Cechy współczesnych systemów operacyjnych

W tablicy 1.1 zestawiono najważniejsze cechy występujące w nowoczesnych systemach operacyjnych.

**Tablica 1.1 Najważniejsze cechy systemów operacyjnych**

Cecha	Wyjaśnienie
<b>wieloprogramowość</b> (ang. multiprogramming)	Możliwość jednoczesnego uruchamiania i przechowywania w pamięci operacyjnej wielu procesów. Przełączanie pomiędzy procesami następuje, gdy proces wykonywany zakończy się lub zamówi operację wejścia - wyjścia.
<b>wielozadaniowość</b> (ang. multitasking) podział czasu (ang. time-sharing)	Możliwość współbieżnego wykonywania wielu procesów z podziałem czasu procesora. Przełączanie pomiędzy poszczególnymi zadaniami odbywa się w regularnych odstępach czasu, na tyle często, że wszyscy użytkownicy mają możliwość pracy interaktywnej.
<b>wieloprzetwarzanie</b>	Możliwość współbieżnego wykonywania procesów przez wiele

(ang. multiprocessing) <b>przetwarzanie równoległe</b> (ang. parallel processing)	procesorów w jednym systemie komputerowym. W wieloprocesorowym systemie komputerowym, czyli ściśle powiązanym, procesory współdzielą pamięć, zegar i szyny systemowe.
<b>wieloprzetwarzanie symetryczne</b> (ang. symmetric multiprocessing - SMP) <b>symetryczne przetwarzanie wieloprocessorowe</b>	Występuje w sytuacji, gdy żaden procesor nie jest wyróżniony i wszystkie mogą wykonywać takie same zadania.
<b>wieloprzetwarzanie asymetryczne</b> (ang. asymmetric multiprocessing)	Występuje w sytuacji, gdy wyróżniony procesor główny zarządza całym systemem i przydziela zadania innym procesorom.
<b>przetwarzanie w czasie rzeczywistym</b> (ang. real-time processing)	Możliwość wykonywania procesów przy ściśle określonych ograniczeniach czasowych. <b>Rygorystyczny system czasu rzeczywistego</b> zapewnia wypełnianie krytycznych zadań (procesów) w gwarantowanym czasie. <b>Łagodny system czasu rzeczywistego</b> zapewnia krytycznym procesom jedynie najwyższy priorytet wykonania, ale nie gwarantuje czasu wykonania.
<b>przetwarzanie rozproszone</b> (ang. distributed processing)	Możliwość współbieżnego wykonywania procesów przez wiele procesorów w systemie rozproszonym. W rozproszonym systemie komputerowym, czyli luźno powiązanym, procesory nie dzielą pamięci ani zegara. Każdy procesor dysponuje własną pamięcią lokalną i może komunikować się z innymi procesorami poprzez sieć komunikacyjną.
<b>pamięć wirtualna</b> (ang. virtual memory)	Możliwość wykonywania procesów, które nie są w całości przechowywane w pamięci operacyjnej systemu. Dzięki temu rozmiar procesu może przekraczać rozmiar fizycznej pamięci operacyjnej.

## Klasyfikacja systemów operacyjnych

Systemy operacyjne można podzielić najogólniej na dwie kategorie:

- systemy specjalnego przeznaczenia,
- systemy ogólnego przeznaczenia.

Pierwsza kategoria obejmuje systemy opracowane specjalnie do wykonywania jakiegoś konkretnego zadania np.:

- systemy czasu rzeczywistego rygorystyczne: sterowanie procesami przemysłowymi, robotami, eksperymentami naukowymi, systemy medyczne, wojskowe,
- systemy przetwarzania transakcji: systemy bankowe, systemy rezerwacji biletów
- systemy wbudowane (ang. embedded systems): zegarki, aparaty fotograficzne, telefony, smartfony, palmtopy, odtwarzacze MP3, systemy nawigacji satelitarnej itp.

Takie systemy są wyspecjalizowanymi programami i często nie są nawet uważane za systemy operacyjne pomimo, że bezpośrednio sprawują pełną kontrolę nad sprzętem.

Systemy ogólnego przeznaczenia pozwalają wykorzystywać system komputerowy do różnych celów w zależności od potrzeb użytkowników. Konkretnie zastosowanie wymaga użycia specjalnego oprogramowania. Można wśród nich wyróżnić:

- systemy jednostanowiskowe,
- systemy wsadowe,
- systemy wielodostępne,
- systemy czasu rzeczywistego łagodne.

Nowoczesne systemy operacyjne są w większości systemami wielodostępnymi, które pozwalają pracować jednocześnie wielu użytkownikom za pośrednictwem terminali podłączonych bezpośrednio do komputera lub poprzez sieć komputerową. Cechę tę posiadają takie systemy, jak: Unix, Linux, Windows 7, MacOS X.

## 1.2. Działanie systemu operacyjnego

Każdy system operacyjny jest po prostu programem lub zbiorem programów komputerowych, które są uruchamiane podobnie jak wszystkie inne programy w systemie komputerowym. Inne jest tylko jego przeznaczenie. Kod systemu operacyjnego wykonywany jest przez procesor (lub wiele procesorów) naprzemiennie z kodem programów użytkowych. Może się to odbywać szeregowo lub równolegle w zależności od zasobów sprzętowych. System nie sprawuje więc stałej kontroli nad systemem komputerowym, a jedynie okresowo przejmuje sterowanie, aby wykonać niezbędne zadania systemowe.

### Przerwania

Współczesny system operacyjny jest sterowany przerwaniami (ang. *interrupt driven*). Mogą one pochodzić z następujących źródeł:

- przerwania z urządzeń wejścia-wyjścia - informują o zakończeniu operacji wejścia-wyjścia,
- przerwania z czasomierza - informują o upływie czasu przydzielonego na wykonywanie programu,
- przerwania programowe (pułapki) – informują o błędach w kodzie wykonywanego programu.

Przerwania z czasomierza przychodzą do procesora w regularnych odstępach czasu, dając w ten sposób możliwość wstrzymania wykonywania bieżącego programu i przejęcia sterowania przez system operacyjny. W ten sposób realizowana jest ochrona CPU przed nieskończonymi pętlami. Po przejęciu sterownia system może wykonywać swoje bieżące zadania takie jak szeregowanie procesów.

Obsługa przerwania przez system operacyjny wygląda następująco:

- zachowanie stanu procesora, czyli zawartości licznika rozkazów, rejestru stanu oraz pozostałych rejestrów procesora,
- określenie typu przerwania poprzez odpytywanie kolejnych urządzeń lub poprzez wektor przerw,
- wykonanie odpowiedniego fragmentu kodu procedury obsługi przerwania.

### Dualny tryb pracy

Współczesne systemy operacyjne wykorzystują **dualny tryb pracy**. System może pracować w jednym z dwóch trybów:

1. tryb systemu (monitora, jądra),
2. tryb użytkownika.

**Tryb systemu**, nazywany również trybem monitora lub jądra, jest trybem uprzywilejowanym. Wykonywany jest w nim kod systemu operacyjnego, a zatem wszystkie operacje istotne dla poprawnego działania systemu komputerowego. W **trybie użytkownika** wykonywany jest kod programów użytkowników. Jednakże każde wystąpienie przerwania powoduje przełączenie trybu pracy na tryb systemu, w którym następuje obsługa przerwania przez system operacyjny, a następnie powrót do trybu użytkownika.

W architekturze sprzętu komputerowego (architekturze procesora) potrzebny jest dodatkowy bit nazywany **bitem trybu**. Jego stan określa aktualny tryb pracy i pozwala odróżnić operacje wykonywane przez system od operacji wykonywanych przez programy użytkowe. Zastosowanie dualnego trybu pracy uniemożliwia programom bezpośredni dostęp do zasobów systemu operacyjnego oraz zasobów innych użytkowników. W ten sposób zapewnia ochronę systemu przed użytkownikami oraz ochronę użytkowników wzajemnie przed sobą.

Programy użytkowników wykonywane są w trybie użytkownika. Jednakże, system operacyjny wyróżnia zbiór **rozkazów uprzywilejowanych**, które mogą być potencjalnie niebezpieczne, np.

operacje na urządzeniach wejścia-wyjścia. Rozkazy takie mogą być wykonywane tylko w trybie systemu, więc programy użytkowników nie mogą ich wykonywać bezpośrednio. Mogą jedynie zlecać ich wykonanie systemowi operacyjnemu za pośrednictwem wywołań systemowych (funkcji systemowych). Przed wykonaniem każdego rozkazu uprzywilejowanego sprzęt komputerowy musi przełączyć tryb pracy na tryb systemu.

### **Usługi systemu operacyjnego**

System operacyjny świadczy różnorodne usługi użytkownikom oraz uruchamianym przez nich programom. Są to przede wszystkim:

- interfejs użytkownika - graficzny GUI (ang. Graphical User Interface) lub tekstowy CLI (ang. Command Line Interface),
- wykonywanie programów,
- realizacja operacji wejścia-wyjścia,
- manipulowanie systemem plików,
- komunikacja,
- wykrywanie błędów w działaniu sprzętu komputerowego oraz programów.

Niektóre usługi mają znaczenie dla działania całego systemu komputerowego:

- przydzielanie zasobów,
- rozliczanie czasu (i ewentualnie kosztów) pracy programów i użytkowników,
- ochrona.

### **Interfejsy programowe**

System operacyjny udostępnia dwa interfejsy programowe:

1. interfejs funkcji systemowych,
2. interfejs programów systemowych.

Funkcje systemowe tworzą interfejs między programami a jądrem systemu operacyjnego. Umożliwiają programom korzystanie z usług jądra i sprzętu komputerowego bez naruszania bezpieczeństwa systemu

Interfejs programów systemowych jest interfejsem użytkownika. Umożliwiają one użytkownikom wykonywanie typowych operacji dotyczących manipulowania plikami, przetwarzania ich zawartości, tworzenia i wykonywania programów, komunikacji czy informowania o stanie systemu. Podstawowe

## 1.3. Budowa systemów operacyjnych

### Składniki systemu operacyjnego

Każdy system operacyjny realizuje wiele zadań. Najważniejsze składniki systemu odpowiadają za:

- zarządzanie procesami (zadaniami),
- zarządzanie pamięcią operacyjną,
- zarządzanie plikami,
- zarządzanie systemem wejścia-wyjścia,
- zarządzanie pamięcią pomocniczą,
- pracę sieciową,
- system ochrony,
- system interpretacji poleceń.

### Struktura systemu

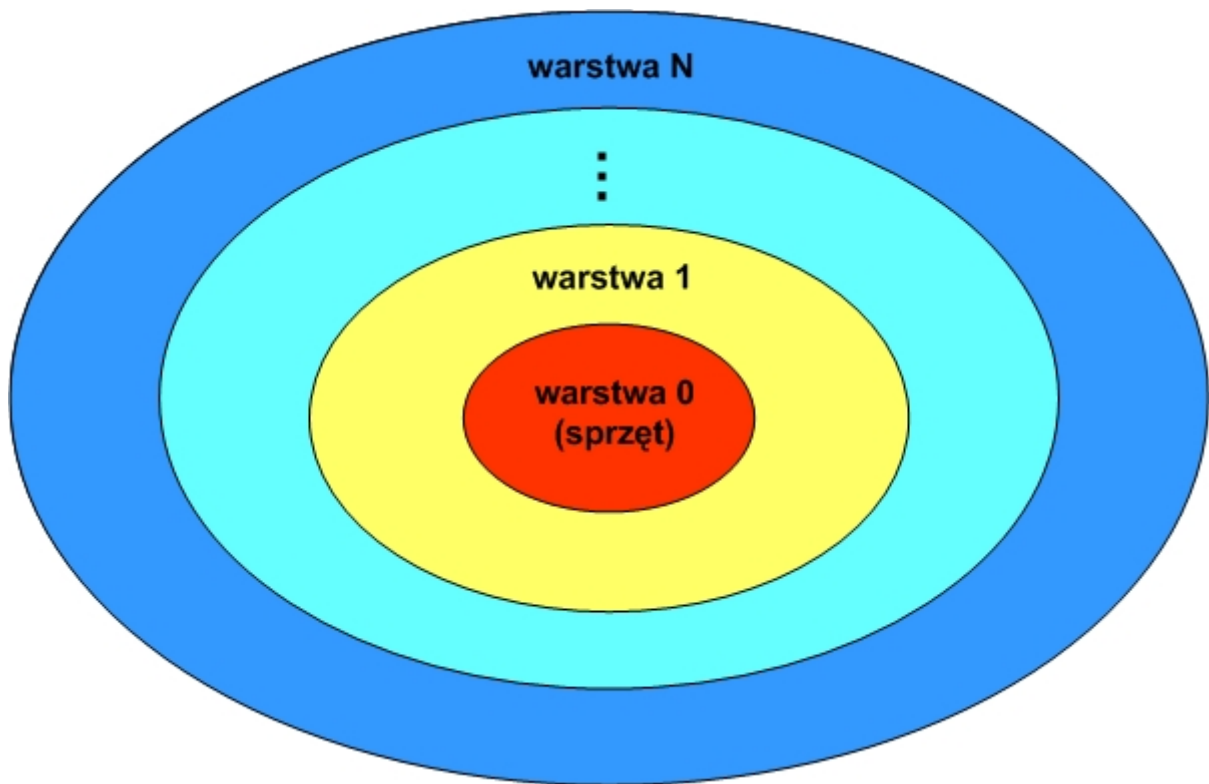
Wyróżnia się dwa zasadnicze typy struktury wewnętrznej systemów operacyjnych:

1. struktura jednolita,
2. struktura modułowa.

System o **strukturze jednolitej** (prostej) stanowi zbiór procedur w ramach jednego programu. Na ogół charakteryzują go słabo wydzielone poziomy funkcjonalne lub w ogóle brak jakiegokolwiek struktury wewnętrznej. Strukturę jednolitą posiadały proste systemy jedno stanowiskowe, takie jak MS-DOS.

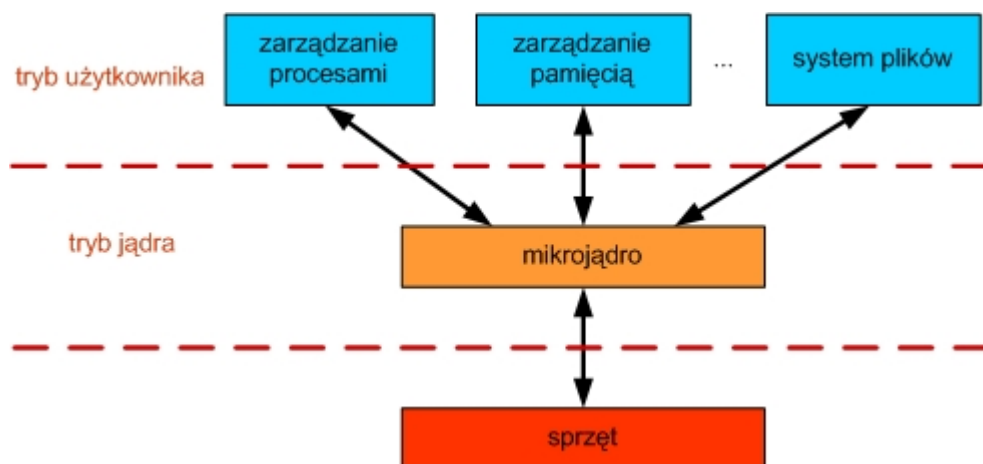
W innym podejściu, wymienione wcześniej typowe składniki systemu operacyjnego mogą być realizowane w postaci oddzielnych modułów w złożonej strukturze systemu operacyjnego. Struktura modułowa zmniejsza stopień zależności między poszczególnymi składowymi systemu oraz ułatwia weryfikację systemu i wyszukiwanie błędów.

W przypadku **struktury warstwowej** system jest podzielony na moduły powiązane w postaci kolejnych warstw jak na Rys. 1.1. Każda warstwa jest zdefiniowana na szczycie poprzedniej warstwy i korzysta z funkcji warstw leżących poniżej. Ukrywa przy tym cechy sprzętu, operacje, struktury danych przed wyższymi warstwami. Podstawowym problemem jest tu trudność definiowania warstw w taki sposób, aby rozdzielić poszczególne funkcje systemu operacyjne na oddzielne warstwy. Daleko posuniętą warstwowość struktury wewnętrznej wykazywały właściwie tylko niektóre systemy akademickie (np. THE, Venus), tworzone w celu zilustrowania tej idei. Ograniczoną strukturę warstwową miały również systemy VMS, OS/2 oraz system UNIX.



Rys. 1.1. Struktura warstwowa systemu operacyjnego

Innym typem struktury modułowej jest **struktura oparta na mikrojądrze**, przedstawiona na Rys. 1.2. Jest to rozwiązanie oparte na modelu klient-serwer. Wyróżniony moduł centralny, określany jako **mikrojądro**, działa w trybie chronionym (trybie jądra) i realizuje tylko kilka podstawowych funkcji systemu. Pozostałe, równorzędne moduły realizują większość usług systemowych, działając w trybie użytkownika. Moduły komunikują się ze sobą za pomocą komunikatów poprzez mikrojądro. Struktura taka odznacza się dużą elastycznością, jest łatwo rozszerzalna o kolejne moduły. Jest też łatwiejsza do zaimplementowania i przenoszenia na nowe platformy sprzętowe. Ze względu na to, że tylko niewielka część kodu działa w trybie jądra, wydaje się być również bardziej niezawodna i bezpieczniejsza. Podstawową wadą okazała się jednak zdecydowanie mniejsza wydajność z powodu używania systemu komunikatów i częstego przełączania pomiędzy trybem użytkownika a trybem jądra. Z tego względu, współczesne systemy operacyjne wykorzystują zmodyfikowaną strukturę z mikrojądrem, w której więcej modułów działa w trybie jądra. Pokażemy to w wykładzie 2 na przykładzie rodziny systemów Windows NT (struktura zmodyfikowana począwszy od Windows NT 4.0) oraz MacOS X.

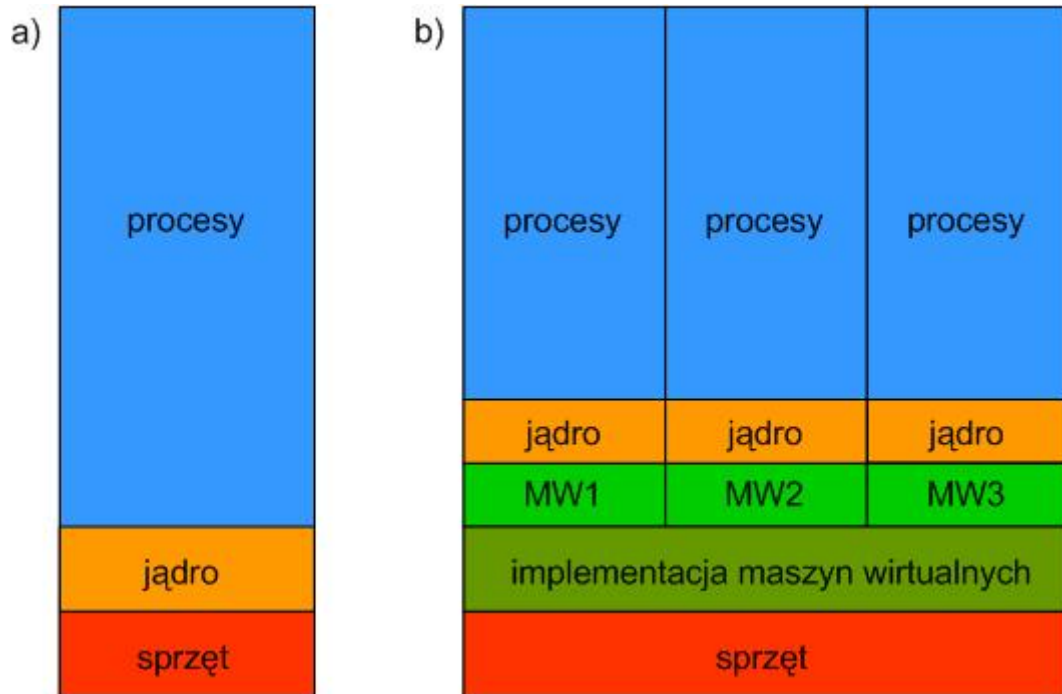


Rys.1.2 Struktura modułowa oparta na mikrojądrze



## Maszyny wirtualne

Rozwinięcie idei warstwowej struktury systemu operacyjnego stanowi koncepcja maszyny wirtualnej zilustrowana na Rys. 1.3. **Maszyna wirtualna** to obraz maszyny rzeczywistej, czyli wirtualna kopia komputera, którą można udostępnić użytkownikowi. Maszyna wirtualna tworzy interfejs identyczny z podstawowym sprzętem komputerowym i ukrywa przed użytkownikiem cechy sprzętu.



Rys. 1.3. Maszyny wirtualne: a) maszyna rzeczywista, b) maszyny wirtualne

## Bibliografia

### Bibliografia podstawowa

1. Wnuk P.: Architektura systemów komputerowych, Ośrodek Kształcenia Na Odległość OKNO, Politechnika Warszawska 2010,
2. Silberschatz A., Galvin P., Gagne G.: Podstawy systemów operacyjnych, WNT 2006 (rozdziały: 1, 3, 22).

### Bibliografia uzupełniająca

1. Główny serwer projektu Linux : [www.linux.org](http://www.linux.org)
2. Linux Documentation Project: [www.tldp.org](http://www.tldp.org)
3. Serwis mający na celu popularyzację systemu Linux: [www.linux.pl](http://www.linux.pl)
4. Serwery polskich Grup Użytkowników Linux-a (Linux User Groups)
  - [www.linux.org.pl](http://www.linux.org.pl)
  - [www.linux.bydgoszcz.pl](http://www.linux.bydgoszcz.pl)
  - [www.silesia.linux.org.pl](http://www.silesia.linux.org.pl)
  - [www.linux.gda.pl](http://www.linux.gda.pl)

## Słownik - nowe pojęcia, definicje i wyrażenia

Termin	Objaśnienie
<b>funkcje systemowe</b>	stanowią interfejs pomiędzy programami a jądrem systemu operacyjnego
<b>logowanie</b>	proces zgłaszania i autoryzacji użytkownika
<b>pamięć wirtualna</b>	możliwość wykonywania procesów, które nie są w całości przechowywane w pamięci operacyjnej systemu
<b>przetwarzanie rozproszone</b>	możliwość współbieżnego wykonywania procesów przez wiele procesorów w systemie rozproszonym
<b>przetwarzanie w czasie rzeczywistym</b>	możliwość wykonywania procesów przy ściśle określonych ograniczeniach czasowych
<b>system czasu rzeczywistego łagodny</b>	system, który zapewnia krytycznym procesom jedynie najwyższy priorytet wykonania, ale nie gwarantuje czasu wykonania
<b>system czasu rzeczywistego rygorystyczny</b>	system, który zapewnia wypełnianie krytycznych zadań (procesów) w gwarantowanym czasie
<b>system wielodostępny</b>	system, który pozwala pracować jednocześnie wielu użytkownikom za pośrednictwem terminali podłączonych bezpośrednio do komputera lub poprzez sieć komputerową
<b>wieloprogramowość</b>	możliwość jednoczesnego uruchamiania i przechowywania w pamięci operacyjnej wielu procesów
<b>wieloprzetwarzanie</b>	możliwość współbieżnego wykonywania procesów przez wiele procesorów w jednym systemie komputerowym
<b>wieloprzetwarzanie asymetryczne</b>	występuje w sytuacji, gdy wyróżniony procesor główny zarządza całym systemem i przydziela zadania innym procesorom
<b>wieloprzetwarzanie symetryczne</b>	występuje w sytuacji, gdy żaden procesor nie jest wyróżniony i wszystkie mogą wykonywać takie same zadania
<b>wielozadaniowość</b>	możliwość współbieżnego wykonywania wielu procesów z podziałem czasu procesora