

# Wprowadzenie do systemów operacyjnych

dr hab. inż. Krzysztof Patan, prof. PWSZ

Instytut Politechniczny  
Państwowa Wyższa Szkoła Zawodowa w Głogowie  
`k.patan@issi.uz.zgora.pl`

# Zadania systemów operacyjnych

## Systemem operacyjnym

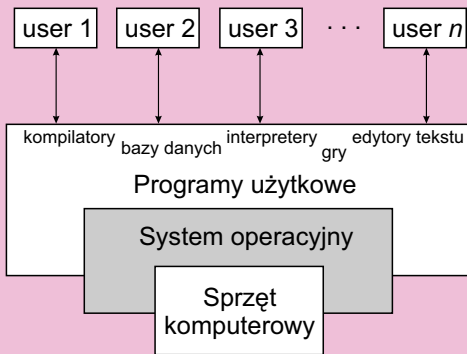
System operacyjny jest programem za pomocą którego użytkownik może korzystać ze sprzętu komputerowego

## Zadanie systemu operacyjnego

Stworzenie takiego środowiska, w którym użytkownik może wykonywać programy w sposób dla niego wygodny, a co najważniejsze wydajny

- system operacyjny (ang. *operating system*) jest ważną częścią składową prawie każdego systemu komputerowego
- w ramach systemu komputerowego wyróżniamy:
  - ◇ sprzęt
  - ◇ system operacyjny
  - ◇ programy użytkowe
  - ◇ użytkowników

## Struktura systemu komputerowego



- Sprzęt komputerowy (ang. *hardware*)
  - ◇ podstawowe składowe:
    - procesor CPU (ang. *central processing unit*)
    - pamięć (ang. *memory*)
    - urządzenia wejścia–wyjścia (ang. *IO devices*)
- Programy użytkowe
  - ◇ określają sposoby i cele użycia sprzętu komputerowego
- Użytkownicy
  - ◇ użytkownicy – ludzie, maszyny, inne komputery
  - ◇ wykorzystują system komputerowy do zaspokojenia własnych potrzeb lub w celu wykonania określonych zadań

# Cele systemu operacyjnego

- System operacyjny

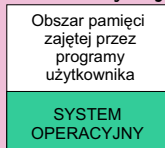
- ◇ nadzoruje i koordynuje posługiwanie się sprzętem komputerowym przez różne programy użytkowe, które pracują dla użytkownika
- ◇ tworzy środowisko (ang. *environment*), w którym inne programy mogą wykonywać pożyteczne prace
- ◇ rozdziela zasoby systemu komputerowego
- ◇ system operacyjny powinien zarządzać zasobami tak aby nie powstawały konflikty
- ◇ nadzoruje działanie programów użytkowych
- ◇ przeciwdziała powstawaniu błędów
- ◇ zapobiega niewłaściwemu użyciu komputera
- ◇ systemy operacyjne powinny być tak skonstruowane aby:
  - ❶ zapewnić użytkownikowi wygodę pracy z komputerem
  - ❷ zapewnić efektywne działanie systemu komputerowego

# Klasyfikacja systemów operacyjnych

- ◇ Proste systemy wsadowe
- ◇ Wieloprogramowe systemy wsadowe
- ◇ Systemy z podziałem czasu
- ◇ Systemy równoległe
- ◇ Systemy rozproszone
- ◇ Systemy sieciowe
- ◇ Systemy czasu rzeczywistego
- ◇ Systemy dla komputerów osobistych

# Proste systemy wsadowe

- ◇ systemy operacyjne pierwszych komputerów obsługiwanych za pomocą konsoli, z czytnikami kart perforowanych
- ◇ komputery – bardzo duże maszyny zajmujące całe pomieszczenia
- ◇ zasada działania – automatyczne przekazywanie sterowania od jednego zadania do następnego
- ◇ system operacyjny na stałe rezyduje w pamięci operacyjnej



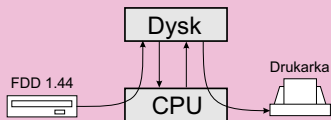
- ◇ przyspieszenie przetwarzania – zadania o podobnych wymaganiach grupowano razem – **wsad**(ang. *batch*)

## Właściwości systemu wsadowego

- ◇ brak bezpośredniego nadzoru ze strony użytkownika podczas wykonywania zadania
- ◇ duża zwłoka między przedłożeniem zadania, a jego zakończeniem (czas obiegu zadania)
- ◇ czas obiegu zadania wynika z:
  - ilości obliczeń
  - opóźnienia rozpoczęcia zadania przez system
- ◇ jednostka centralna często pozostawała bezczynna
- ◇ szybkość działania urządzeń mechanicznych dużo mniejsza od urządzeń elektronicznych (3 i więcej)
  - ◇ **rozwiązanie** – polepszenie wydajności urządzeń wejścia–wyjścia
  - ◇ wraz z polepszeniem wydajności urządzeń wejścia–wyjścia, jeszcze bardziej wzrosła wydajność procesorów – dalsze pogłębienie problemu

## Spooling

- ◇ zadania zamiast w pamięci są przechowywane na dysku
- ◇ rozmieszczenie zadań na dysku jest pamiętane w tablicy znajdującej się w pamięci operacyjnej
- ◇ **SPOOL** - jednoczesna bezpośrednia praca urządzeń (ang. *Simultaneous Peripheral Operation On-Line*)



## Spooling – cd.

- ◇ używa się dysku jako olbrzymiego bufora do czytania danych z maksymalnym wyprzedzeniem z urządzeń wejściowych lub do przechowywania plików wyjściowych do czasu aż urządzenia wyjściowe będą w stanie je przyjąć
- ◇ metoda umożliwia nakładanie w czasie operacji wejścia–wyjścia jednego zadania na obliczenia przez procesor innych zadań
- ◇ utrzymanie procesora i urządzeń wejścia–wyjścia w większej aktywności

# Wieloprogramowe systemy wsadowe

- ◇ spooling powoduje, że pewna liczba zadań jest zawczasu czytana na dysk, gdzie czeka gotowa do wykonania (pula zadań)
- ◇ zadanie z puli wybiera się tak, aby zwiększyć wykorzystanie CPU
- ◇ planowanie zadań (szeregowanie zadań)
- ◇ dzięki istnieniu puli zadań możliwe jest wieloprogramowanie



## Właściwości

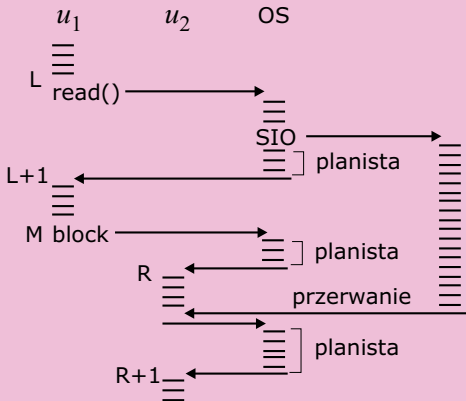
- ◇ systemy wieloprogramowe decydują za użytkowników, które procesy mają zostać wykonane
- ◇ dopóki są jakieś zadania w pamięci do wykonania jednostka centralna nie jest bezczynna
- ◇ jeśli nie wszystkie zadania z puli zadań mogą zostać umieszczone w pamięci to system operacyjny musi wybierać spośród nich
- ◇ przechowywanie wielu zadań w pamięci wymaga mechanizmów zarządzania pamięcią
- ◇ jeśli kilka zadań w pamięci operacyjnej jest gotowych do działania to należy wybrać któryś z nich i przydzielić mu procesor
- ◇ systemy tego typu dużo są bardziej skomplikowane od prostych systemów wsadowych

## Ilustracja pracy systemu wielozadaniowego

RAM

|       |
|-------|
| OS    |
| $u_1$ |
| $u_2$ |
| $u_3$ |
| $u_4$ |

CPU



# Systemy z podziałem czasu

- ◇ systemy wieloprogramowe tworzą środowisko, w którym zasoby systemowe są skutecznie użytkowane
- ◇ system wieloprogramowy jest kłopotliwy z punktu widzenia użytkownika
  - nie można ingerować w program podczas jego wykonania
  - należy przygotować procedury sterujące na wypadek wszystkich możliwych zdarzeń
  - programista nie może na bieżąco zmieniać programu, aby zaobserwować jego zachowanie

## Właściwości

### **Podział czasu** – wielozadaniowość, multitasking

- logiczne rozszerzenie wieloprogramowości
- procesor wykonuje wiele zadań, ale przełącza się od jednego do następnego
- przełączenia następują tak często, że użytkownicy mogą współdziałać z każdym programem podczas jego wykonania

## Właściwości – cd.

- system z podziałem czasu jest systemem interakcyjnym
- zadania interakcyjne składają się z wielu krótkich działań (w przeciwieństwie do wielkich zadań wsadowych)
- w systemach z podziałem czasu zastosowano planowanie przydziału czasu procesora i wieloprogramowość; umożliwia to każdemu użytkownikowi możliwość korzystania z małej porcji dzielonego czasu pracy komputera
- wielu użytkowników dzieli równocześnie jeden komputer
- dzięki błyskawicznym przełączeniom od jednego użytkownika do drugiego każdy z nich ma wrażenie, że dysponuje własnym komputerem
- jest to system jeszcze bardziej złożony od systemów wieloprogramowych

# Systemy równoległe

- ◇ systemy komputerowe są w większości jednoprocessorowe
- ◇ duże zainteresowanie poświęca się systemom wieloprocessorowym
- ◇ w systemach wieloprocessorowych pewna liczba procesorów współpracuje ze sobą, dzielą magistrale komputera, zegar, pamięć i urządzenia zewnętrzne
- ◇ systemy równoległe nazywane są ściśle powiązanymi
- ◇ zwiększając liczbę procesorów można oczekiwać, że większą ilość pracy da się wykonać w krótszym czasie; jaki jest współczynnik przyspieszenia przy  $n$  procesorach?

## Zalety

- zwiększenie przepustowości
- niezawodność
- wspólne użytkowanie przez wiele procesorów urządzeń

## Podział systemów równoległych

- przetwarzanie symetryczne (ang. *simultaneous multiprocessing*)
  - ◊ każdy procesor pracuje pod identyczną kopią systemu operacyjnego
  - ◊ można uruchomić wiele procesów jednocześnie

### Przykłady:

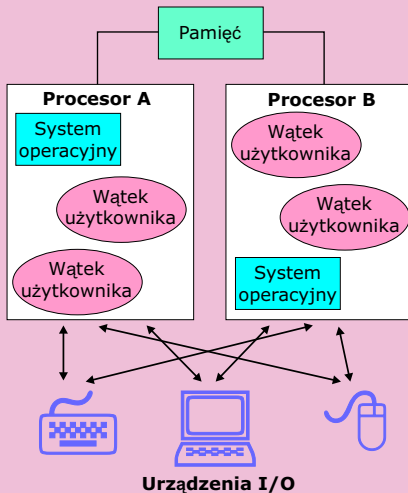
Windows NT/2000, OS/2 Warp, IBM PC Server 320, 520, 720

- przetwarzanie asymetryczne (ang. *asymmetric multiprocessing*)
  - ◊ każdy procesor wykonuje ściśle określone zadania
  - ◊ w systemie istnieje jeden procesor nadrzędny, który rozdziela i przydziela zadania do procesorów podrzędnych
  - ◊ przetwarzanie tego typu jest stosowane w bardzo dużych systemach komputerowych

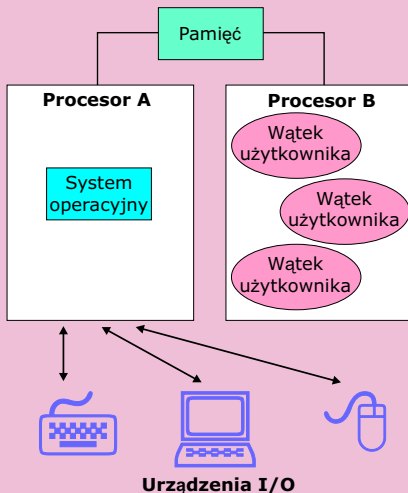
### Przykłady:

VM/370, PS/2 Server 195 i 295

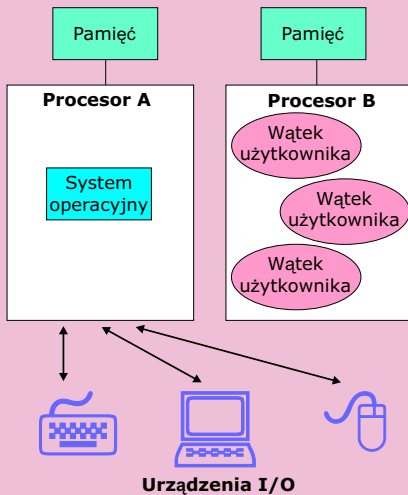
## Ilustracja zasady działania wieloprzetwarzania symetrycznego



## Ilustracja zasady działania wieloprzetwarzania asymetrycznego – przypadek 1



## Ilustracja zasady działania wieloprzetwarzania asymetrycznego – przypadek 2



# Systemy sieciowe

Sieciowy system operacyjny (ang. *network operating system*) tworzy środowisko, w którym użytkownicy, świadomi wielości maszyn, mają dostęp do zasobów zdalnych

- ◇ W systemie sieciowym procesory nie dzielą pamięci ani zegara; każdy procesor posiada własną pamięć i zegar
- ◇ Komputery komunikują się za pomocą różnych linii komunikacyjnych
- ◇ Połączenia komputerów określa topologia sieci: magistrala, pierścień, gwiazda, hierarchiczna, każdy-z-każdym, itd.
- ◇ W celu dostępu do zasobów sieciowych należy rejestrować się na maszynie zdalnej
- ◇ Użytkownik pracujący na maszynie lokalnej (np. Windows NT) rejestrując się na maszynie zdalnej (np. UNIX) musi na czas trwania sesji stosować polecenia systemu zdalnego (UNIX)

## Usługi systemów sieciowych

- Współdzielenie zasobów i plików
  - ◇ drogie urządzenia (drukarki, plotery) można współdzielić
  - ◇ możliwość współdzielenia zasobów plikowych (bazy danych)
- Komunikacja
  - ◇ możliwość wymiany informacji
  - ◇ przeglądanie plików, obsługa poczty elektronicznej, wywoływanie procedur zdalnych
  - ◇ działania można wywoływać na wielkie odległości
  - ◇ koordynacja pracy zespołów badawczego, roboczego

### Przykłady systemów sieciowych:

Novell NetWare, Windows NT Server, Windows 2000 Server/Advanced Server, UNIX/Linux

# Systemy rozproszone

System rozproszony (ang. *distributed system*) jest zbiorem samodzielnych komputerów połączonych za pomocą sieci i wyposażonych w rozproszone oprogramowanie systemowe. Użytkownicy systemu rozproszonego odbierają go jako jedno, zintegrowane środowisko obliczeniowe

- ◇ Tendencja do rozdzielania obliczeń między wiele procesorów
- ◇ Użytkownicy nie muszą być świadomi wielości maszyn
- ◇ Dostęp do zasobów zdalnych uzyskuje się tak samo jak do zasobów lokalnych
- ◇ Systemy rozproszone nazywane są luźno powiązanymi

## Właściwości

### ❶ Dzielenie zasobów

- wspólne korzystanie z plików na zdalnych stanowiskach
- przetwarzanie informacji w rozproszonych bazach danych
- drukowanie zadań na zdalnych stanowiskach
- używanie wyspecjalizowanych zdalnych urządzeń

### ❷ Przyspieszenie obliczeń

- rozdzielenie obliczeń między różne stanowiska w celu równoległego wykonania
- dzielenie obciążeń – jeżeli dane stanowisko jest przeciążone to część jego zadań można przenieść na inne stanowisko

### ❸ Niezawodność

- w razie awarii danego stanowiska pozostałe kontynuują działanie
- utrzymanie pewnej nadmiarowości stanowisk powoduje utrzymanie efektywności pomimo awarii

## Przykłady systemów rozproszonych

**Mach** – Carnegie-Mellon University (USA) –  
lata 80-te

**Chorus** – INRIA (Francja) – 1979

**Amoeba** – Vrije Universiteit (Holandia) - 1981

# Systemy czasu rzeczywistego

- Systemy czasu rzeczywistego (ang. *real-time systems*) stosuje się tam gdzie istnieją surowe wymagania na czas wykonania operacji czy przepływu danych
- System rygorystyczny, twardy (ang. *hard real-time system*)
  - ◇ gwarantuje terminowe wypełnianie zadań krytycznych
  - ◇ ograniczenie wszystkich opóźnień w systemie
  - ◇ dane są przechowywane w pamięci o krótkim czasie dostępu
  - ◇ pamięć pomocnicza jest mała lub nie istnieje wcale
  - ◇ system kłóci się z systemami z podziałem czasu
  - ◇ zastosowanie w procesach przemysłowych, sterowanie liniami technologicznymi, systemach kontroli lotu, itp.

## Przykłady twardych systemów czasu rzeczywistego:

QNX, RTLinux, LynxOS, VxWorks

- System łagodny (ang. *soft real-time system*)
  - ◇ mniej wymagający system
  - ◇ system umożliwia pracę z systemami innych rodzajów
  - ◇ zastosowanie w przemyśle jest ryzykowne
  - ◇ znalazły zastosowanie w multimediach, wirtualnej rzeczywistości, eksploracjach morskich i planetarnych, systemach wizyjnych

**Przykłady miękkich systemów czasu rzeczywistego:**  
Standardowy system Linux, Windows XP Embedded