

Systemy operacyjne

## Procesy, zasoby i wątki

Wykład prowadzą:  
Jerzy Brzeziński  
Dariusz Wawrzyniak



UCZELNIA  
ONLINE

---

---

---

---

---


---

---

---

Systemy operacyjne

### Plan wykładu



- Koncepcja procesu i zasobu
- Obsługa procesów i zasobów
- Cykl zmian stanów procesu i kolejkowanie
- Klasyfikacja zasobów
- Wątki
- Procesy i wątki we współczesnych systemach operacyjnych

Procesy, zasoby i wątki (2)

---

---

---

---

---


---

---

---

Systemy operacyjne

### Koncepcja procesu



- Proces jest elementarną jednostką pracy (aktywności) zarządzaną przez system operacyjny, która ubiega się o zasoby systemu komputerowego w celu wykonania programu.
- Proces = wykonujący się program.
- Elementy składowe procesu:
  - program — definiuje zachowanie procesu,
  - dane — zbiór wartości przetwarzanych oraz wyniki,
  - zbiór zasobów tworzących środowisko wykonawcze,
  - blok kontrolny procesu (PCB, deskryptor) — opis bieżącego stanu procesu.

Procesy, zasoby i wątki (3)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Koncepcja zasobu**

- Zasobem jest element sprzętowy lub programowy systemu komputerowego, którego brak może potencjalnie zablokować wykonywanie programu (przetwarzanie)
- Przykłady zasobów: procesor, pamięć, plik (dane) itp.

Procesy, zasoby i wątki (4)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Podział operacji jądra systemu w zarządzaniu procesami i zasobami**

- Operacje tworzenia i usuwania procesów oraz elementarna komunikacja międzyprocesowa
- Operacje przydziału i zwalniania jednostek zasobów
- Elementarne operacje wejścia-wyjścia
- Procedury obsługi przerw

Procesy, zasoby i wątki (5)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Zarządcy**

- Zarządca procesów (process manager) — kontroluje stany procesów w celu efektywnego i bezpiecznego wykorzystania współdzielonych zasobów systemu.
- Zarządca zasobów (resource manager) — realizuje przydział zasobów stosownie do żądań procesów, aktualnego stanu systemu oraz ogólnosystemowej polityki przydziału.

Procesy, zasoby i wątki (6)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Struktury danych**

- Deskryptor procesu (blok kontrolny procesu, PCB) — używany przez zarządcę procesów w celu rejestrowania stanu procesu w czasie jego monitorowania i kontroli.
- Deskryptor zasobu — przechowuje informacje o dostępności i zajętości danego typu zasobu.

Procesy, zasoby i wątki (7)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Deskryptor procesu**

- Identyfikator procesu
- Stan procesu (nowy, gotowy, oczekujący itd.)
- Identyfikator właściciela
- Identyfikator przodka
- Lista przydzielonych zasobów
- Zawartość rejestrów procesora
- Prawa dostępu (domena ochrony)
- Informacje na potrzeby zarządzania pamięcią
- Informacje na potrzeby planowania (np. priorytet)
- Informacje do rozliczeń
- Wskaźniki do kolejek porządkujących

Procesy, zasoby i wątki (8)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Stany procesu**

- Nowy (ang. new) — proces jest tworzony.
- Wykonywany (ang. running) — wykonywane są instrukcje programu.
- Oczekujący (ang. waiting) — proces oczekuje na jakieś zdarzenie, np. na zakończenie operacji wejścia-wyjścia, na przydział dodatkowego zasobu, synchronizuje się z innymi procesami.
- Gotowy (ang. ready) — proces czeka na przydział procesora.
- Zakończony (ang. terminated) — proces zakończył działanie i zwalnia zasoby.

Procesy, zasoby i wątki (9)

---

---

---

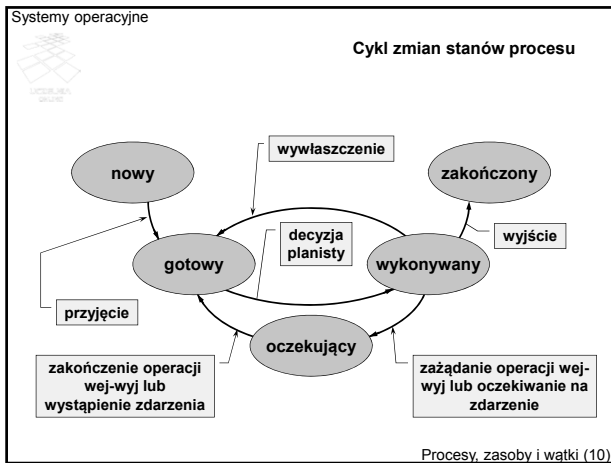
---

---

---

---

---




---

---

---

---

---

---

---

---

- Systemy operacyjne
- Deskryptor zasobu**
- Identyfikator zasobu
  - Rodzaj zasobu
  - Identyfikator twórcy zasobu
  - Lista i liczba dostępnych jednostek zasobu
  - Lista (kolejka) procesów oczekujących na jednostki danego zasobu
  - Procedura przydziału
- Procesy, zasoby i wątki (11)

---

---

---

---

---

---

---

---

- Systemy operacyjne
- Klasyfikacja zasobów**
- Ze względu na sposób wykorzystania
    - zasoby odzyskiwalne (zwrotne, ang. reusable),
    - zasoby nieodzyskiwalne (niezwrotne, zużywalne, ang. consumable).
  - Ze względu na sposób odzyskiwania
    - zasoby wywłaszczalne,
    - zasoby niewywłaszczalne.
  - Ze względu na tryb dostępu
    - zasoby współdzielone
    - zasoby wyłączne
- Procesy, zasoby i wątki (12)

---

---

---

---

---

---

---

---

Systemy operacyjne

### Kolejki procesów

- Kolejka zadań (ang. job queue) — wszystkie procesy systemu.
- Kolejka procesów gotowych (ang. ready queue) — procesy gotowe do działania, przebywające w pamięci głównej.
- Kolejka do urządzenia (ang. device queue) — procesy czekające na zakończenie operacji wejścia-wyjścia.
- Kolejka procesów oczekujących na sygnał synchronizacji od innych procesów (np. kolejka procesów na semaforze).

Procesy, zasoby i wątki (13)

---

---

---

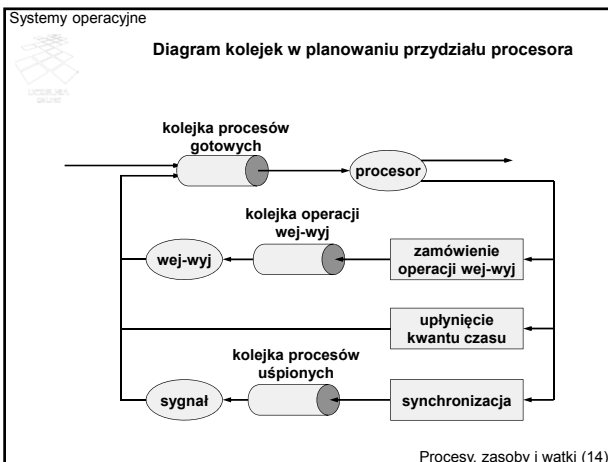
---

---

---

---

---




---

---

---

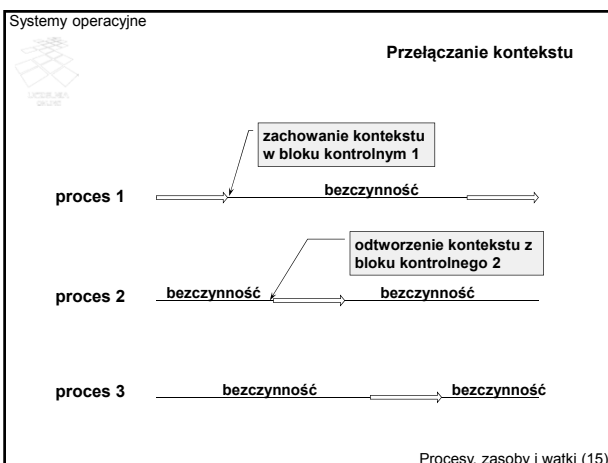
---

---

---

---

---




---

---

---

---

---

---

---

---

Systemy operacyjne

**Planista**

- Planista krótkoterminowy, planista przydziału procesora (ang. CPU scheduler) — zajmuje się przydziałem procesora do procesów gotowych.
- Planista średnioterminowy (ang. medium-term scheduler) — zajmuje się wymianą procesów pomiędzy pamięcią główną a pamięcią zewnętrzną (np. dyskiem).
- Planista długoterminowy, planista zadań (ang. long-term scheduler, job scheduler) — zajmuje się ładowaniem nowych programów do pamięci i kontrolą liczby zadań w systemie oraz ich odpowiednim doбором w celu zrównoważenia wykorzystania zasobów.

Procesy, zasoby i wątki (16)

---

---

---

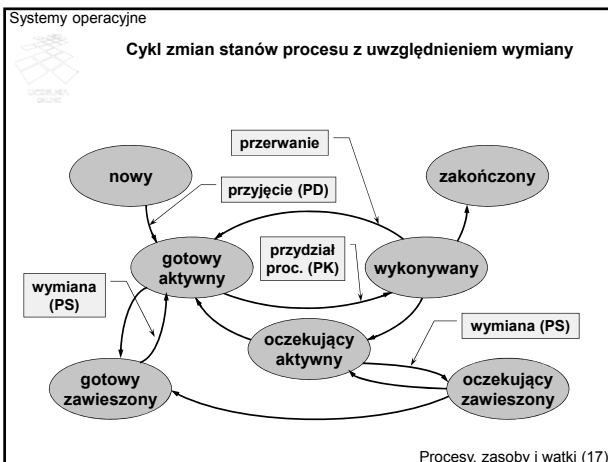
---

---

---

---

---




---

---

---

---

---

---

---

---

Systemy operacyjne

**Obsługa procesów (1)**

- Tworzenie procesu
  - POSIX: `fork`
  - Windows: `CreateProcess`
- Usuwanie procesu
  - POSIX: `exit`, `abort`, `kill`
  - Windows: `ExitProcess`, `TerminateProcess`

Procesy, zasoby i wątki (18)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Obsługa procesów** <sup>(2)</sup>

- Zawieszanie i aktywacja procesu
- Wstrzymywanie i wznowianie procesu
- Zmiana priorytetu procesu
  - POSIX: nice (setpriority)
  - Windows: SetPriorityClass
- Oczekiwanie na zakończenie procesu
  - POSIX: wait, waitpid
  - Windows: brak bezpośredniego wsparcia, należy użyć odpowiednich mechanizmów synchronizacji

Procesy, zasoby i wątki (19)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Elementarne operacje na zasobach**

- Tworzenie deskryptora zasobu
- Usuwanie deskryptora zasobu
- Realizacja żądania przydziału jednostek zasobu
- Zwolnienie i odzyskiwanie jednostek zasobu odzyskiwalnego
- Uwolnienie (wyprodukowanie) jednostki zasobu nieodzyskiwalnego

Procesy, zasoby i wątki (20)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Wątki**

- Wątek (lekki proces, ang. lightweight process — LWP) jest obiektem w obrębie procesu ciężkiego (heavyweight), posiadającym własne sterowanie i współdzielącym z innymi wątkami tego procesu przydzielone (procesowi) zasoby:
  - segment kodu i segment danych w pamięci
  - tablicę otwartych plików
  - tablicę sygnałów

Procesy, zasoby i wątki (21)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Realizacja wątków**

- Realizacja wątków na poziomie jądra systemu operacyjnego — jądro tworzy odpowiednie struktury (blok kontrolny) do utrzymywania stanu wątku.
- Realizacja wątków na poziomie użytkownika — struktury związane ze stanami wątków tworzone są w przestrzeni adresowej procesu.

Procesy, zasoby i wątki (22)

---



---



---



---



---




---



---

Systemy operacyjne



**Realizacja wątków na poziomie jądra systemu operacyjnego**

- Wątek posiada własny blok kontrolny w jądrze systemu operacyjnego, obejmujący:
  - stan licznika rozkazów,
  - stan rejestrów procesora,
  - stan rejestrów związanych z organizacją stosu.
- Własności realizacji wątków na poziomie jądra:
  - przełączanie kontekstu pomiędzy wątkami przez jądro,
  - większy koszt przełączania kontekstu,
  - bardziej sprawiedliwy przydział czasu procesora.

Procesy, zasoby i wątki (23)

---



---



---



---



---




---



---

Systemy operacyjne



**Realizacja wątków w trybie użytkownika**

- Deskryptor wątku znajduje się w tablicy wątków w pamięci danego procesu (jądro nie wie nic o wątkach).
- Własności realizacji na poziomie użytkownika:
  - przydział czasu procesora dla procesu (nie dla wątku)
  - przełączanie kontekstu pomiędzy wątkami przez jawne odwołania do mechanizmu obsługi wątków
  - mniejszy koszt przełączania kontekstu (bez angażowania jądra systemu operacyjnego)
  - możliwość „głodzenia” wątków tego samego procesu, gdy jeden z nich spowoduje przejście procesu w stan oczekiwania

Procesy, zasoby i wątki (24)

---



---



---



---



---

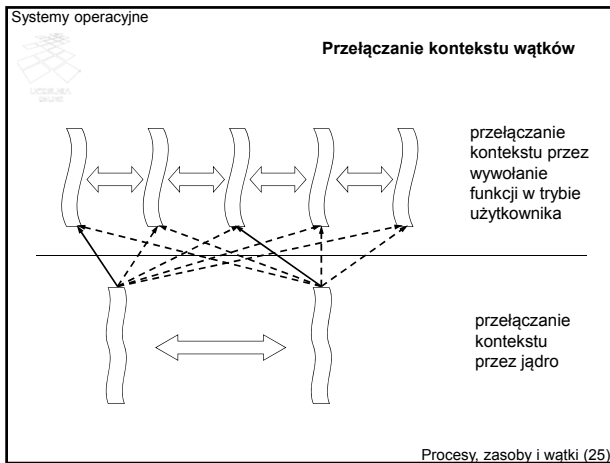


---



---






---

---

---

---

---

---

---

---

- Systemy operacyjne
- ### Obsługa wątków <sup>(1)</sup>
- Tworzenie wątku
    - POSIX: `pthread_create`
    - Windows: `CreateThread`, `CreateRemoteThread`
  - Usuwanie wątku
    - POSIX: `pthread_exit`, `pthread_cancel`
    - Windows: `ExitThread`, `TerminateThread`
- Procesy, zasoby i wątki (26)

---

---

---

---

---

---

---

---

- Systemy operacyjne
- ### Obsługa wątków <sup>(2)</sup>
- Wstrzymywanie i wznowianie wątku
    - POSIX: brak
    - Windows: `SuspendThread`, `ResumeThread`
  - Zmiana priorytetu wątku
    - POSIX: `pthread_setschedprio`, `pthread_setschedparam`
    - Windows: `SetThreadPriority`
  - Oczekiwanie na zakończenie wątku
    - POSIX: `pthread_join`
    - Windows: brak bezpośredniego wsparcia, należy użyć odpowiednich mechanizmów synchronizacji
- Procesy, zasoby i wątki (27)

---

---

---

---

---

---

---

---

Systemy operacyjne

**Realizacja procesów/wątków w systemie Linux**

- W jądrze systemu Linux nie odróżnia się pojęcia wątku od procesu.
- Procesy mogą współdzielić takie zasoby, jak:
  - przestrzeń adresowa,
  - otwarte pliki,
  - informacje o systemie plików,
  - procedury obsługi sygnałów.
- Deskryptory procesów (o strukturze `struct task_struct`) przechowywane są na dwukierunkowej, cyklicznej liście zadań.

Procesy, zasoby i wątki (28)

---

---

---

---

---

---

---

---

Systemy operacyjne

**Stany procesu (wątku) w systemie Linux**

- `TASK_RUNNING` — wykonywanie lub gotowość (do wykonania)
- `TASK_INTERRUPTIBLE` — oczekiwanie na zajście zdarzenia lub sygnał
- `TASK_UNINTERRUPTIBLE` — oczekiwanie na zajście zdarzenia, przy czym sygnały są ignorowane
- `TASK_ZOMBI` — stan zakończenia utrzymywany w celu przechowania deskryptora procesu
- `TASK_STOP` — zatrzymanie w wyniku otrzymania sygnału (np. `SIGSTOP`)

Procesy, zasoby i wątki (29)

---

---

---

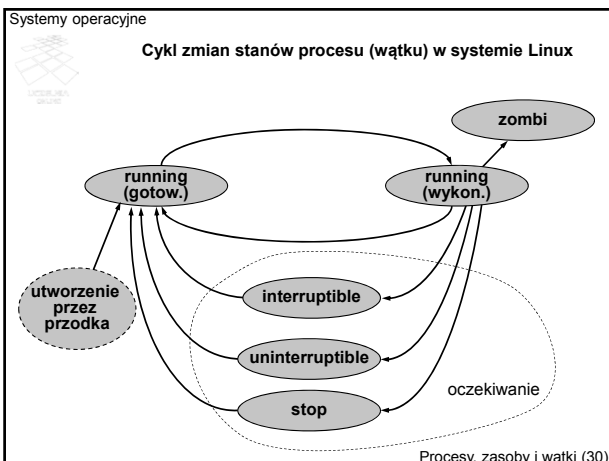
---

---

---

---

---




---

---

---

---


---

---

---

---

Systemy operacyjne



**Proces w systemie Windows 2000/XP**

- Proces stanowi środowisko do wykonywania wątków.
- Struktury opisu procesu obejmują:
  - EPROCESS — blok centrum wykonawczego, opisujący proces,
  - KPROCESS — blok kontrolny procesu, część struktury EPROCESS,
  - PEB — blok środowiska procesu, dostępny w trybie użytkownika.

Procesy, zasoby i wątki (31)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Wątki w systemie Windows 2000/XP**

- Wątki korzystają z zasobów przydzielonych procesom.
- Wątki (nie procesy) ubiegają się o przydział procesora i są szeregowane przez planistę krótkoterminowego.
- Struktury opisu wątku obejmują:
  - ETHREAD — blok centrum wykonawczego, opisujący wątek,
  - KTHREAD — blok kontrolny wątku, część struktury ETHREAD,
  - TEB — blok środowiska wątku, dostępny w trybie użytkownika.

Procesy, zasoby i wątki (32)

---

---

---

---


---

---

---

---

Systemy operacyjne



**Stany wątku w systemie Windows 2000/XP**

- Inicjalizowany (initialized, wartość 0) — stan wewnętrzny w trakcie tworzenia wątku,
- Gotowy (ready, wartość 1) — oczekuje na przydział procesora,
- Wykonywany (running, wartość 2),
- Czuwający (standby, wartość 3) — wybrany do wykonania jako następny,
- Zakończony (terminated, wartość 4),
- Oczekujący (waiting, wart. 5) — oczekuje na zdarzenie,
- Przejście (transition, wartość 6) — oczekuje na sprowadzenie swojego stosu jądra z pliku wymiany,
- Unknown (wart. 7)

Procesy, zasoby i wątki (33)

---

---

---

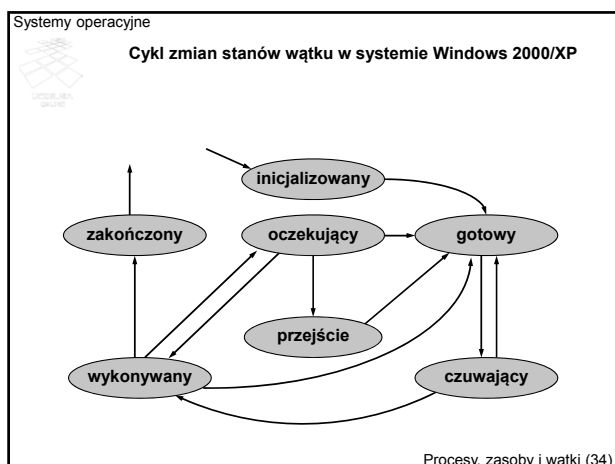
---

---

---

---

---



---

---

---

---

---

---

---

---