

Wątki

Zbigniew Suski

Wątki

1

Wątki - własności

- ❑ Wątek jest podstawową jednostką wykorzystania procesora. Posiada:
 - własny licznik rozkazów,
 - zbiór rejestrów,
 - stos,
 - stan,
 - wątki potomne.
- ❑ Wątek współdzieli z innymi wątkami tego samego zadania:
 - przestrzeń adresową,
 - zmienne globalne,
 - zasoby systemowe.
- ❑ Tradycyjny proces (ciężki) jest równoważny zadaniu z jednym wątkiem.

Zbigniew Suski

Wątki

2

Implementacje wątków

- ❑ W przestrzeni użytkownika (projekt Andrew z CMU)
 - dobra wydajność,
 - potrzebny system wykonawczy wspomagający wątki,
 - możliwość stosowania różnych algorytmów planowania,
 - kłopotliwa realizacja blokowanych odwołań do systemu.
- ❑ W przestrzeni jądra systemu (Mach, OS/2)
 - gorsza wydajność,
 - niepotrzebny system wykonawczy,
 - implementacja nie wymaga „sztuczek”
- ❑ Podejście hybrydowe (Solaris 2)

Zbigniew Suski

Wątki

3

Wątki – podstawowe funkcje

```
#include < pthread.h >
```

<i>pthread_create</i>	tworzenie wątku
<i>pthread_join</i>	oczekiwanie na koniec wątku
<i>pthread_self</i>	pobranie identyfikatora wątku
<i>pthread_detach</i>	odłączenie wątku
<i>pthread_exit</i>	zakończenie wątku

Zbigniew Suski

Wątki

4

Synchronizacja wątków

Zamek (*mutex*): rodzaj ograniczonego semafora binarnego

- typ `pthread_mutex_t`
- `PTHREAD_MUTEX_INITIALIZER`
- funkcja `pthread_mutex_init`
- funkcja `pthread_mutex_destroy`

Stany:

- otwarty
- zamknięty

Operacje:

- zamknij (`pthread_mutex_lock`)
- otwórz (`pthread_mutex_unlock`)
- spróbuj zamknąć (`pthread_mutex_trylock`)

Synchronizacja wątków

Zmienna warunkowa (*condition variable*)

- skojarzona z zamkiem

`pthread_cond_wait`

uśpienie wątku

`pthread_cond_signal`

budzenie wątku

`pthread_cond_broadcast`

budzenie wszystkich wątków

`pthread_cond_timedwait`

definiowanie ograniczenia
czasu zablokowania

Przykłady

plik – [Wątki-przykłady](#)