

Lab 5

1 Tworzenie wątku

1) Przypomnij sobie różnice między procesem a wątkiem:

Wątek – część programu wykonywana współbieżnie w obrębie jednego procesu; w jednym procesie może istnieć wiele wątków.

Różnica polega na współdzieleniu przez wszystkie wątki działające w danym procesie przestrzeni adresowej oraz wszystkich innych struktur systemowych (np. listy otwartych plików, gniazd itp.) – z kolei procesy posiadają niezależne zasoby.

2) `pthread_create(3)`:

tworzy nowy wątek, który wykonuje się współbieżnie z wątkiem wołającym. Nowy wątek zaczyna wykonywać podaną funkcję `start_routine` podając jej `arg` jako pierwszy argument.

2 Czekanie na zakończenie wątku

1) `pthread_join(3)`:

funkcja czeka na zakończenie wątku.

3 Synchronizacja wątków (muteksy)

1) `pthread_mutex_lock` – służy do blokowania muteksa (objekty, które uniemożliwiają wejście do sekcji krytycznej więcej niż jednemu wątkowi)

2) `pthread_mutex_unlock` – do odblokowania

4 Zmienne warunkowe

1) `pthread_cond_signal` – wznawia jeden z wątków oczekujących na zmiennej warunkowej `cond`.

2) `pthread_cond_wait` – zwalnia mutex (tak jak przez `pthread_unlock_mutex`) i oczekuje na sygnał o spełnieniu zmiennej warunkowej `cond`. Wykonywanie wątku jest zawieszane i nie zajmuje czasu procesora aż do odebrania sygnału zmiennej warunkowej.

3) `static pthread_cond_t cond = PTHREAD_COND_INITIALIZER` : tworzenie nowej zmiennej warunkowej

4) `pthread_cond_destroy` – usuwanie zmiennej war.

5 Kasowanie wątku

1) `pthread_exit`

2) `pthread_cancel` – zamyka wątek, kiedy ten będzie się znajdował w punkcie kasowania

3) `pthread_testcancel` – tworzy punkt kasowania w wątku