

Programovanie vo Windows

Cvičenie 3 - Vlákna

Andrea Číková

Martin Osovský

Proces vs. vlákno

Proces

- Objekt jadra
- Adresový priestor
- Nečinný

Vlákno

- Objekt jadra
- Zásobník vlákna
- Vykonáva kód
- Vytvorené v kontexte procesu

Funkcia vstupného bodu vlákna

- Primárne vlákno – `(w)main` alebo `(w)WinMain`
- Ostatné vlákna:

```
DWORD WINAPI ThreadFunc(PVOID param)
{
    DWORD result = 0;
    ...
    return result;
}
```

Vytvorenie vlákna

- Funkcia CreateThread:

```
HANDLE CreateThread(  
    PSECURITY_ATTRIBUTES threadAttr,  
    DWORD stackSize,  
    PTHREAD_START_ROUTINE startAddress,  
    PVOID param,  
    DWORD initFlags,  
    PDWORD threadID);
```

- Pri volaní funkcií C run-time knižnice použiť miesto CreateThread funkciu `_beginthreadex`

Ukončenie vlákna

- Funkcia vlákna skončí
- `ExitThread` (`_endthreadex`)
- `TerminateThread`
- Proces obsahujúci vlákno skončí

Ďalšie užitočné funkcie

```
DWORD ResumeThread(HANDLE hThread);  
DWORD SuspendThread(HANDLE hThread);  
VOID Sleep(DWORD milliseconds);
```

Synchronizácia vlákien

- V užívateľskom režime:
 - Interlocked* funkcie
 - kritické sekcie
- Pomocou objektov jadra:
 - „Dispatcher object“ – akýkoľvek objekt jadra, na ktorý môžete čakať, typicky event, mutex, semaphore
 - Funkcie WaitForSingleObject a WaitForMultipleObjects

Wait funkcie - detaily

- `WaitForSingleObject ()`:
 - `hObject` špecifikuje objekt jadra
 - `milliseconds` - čas v ms
 - 0 – nečaká, skontroluje, či je objekt signalizovaný
 - `INFINITE` – čaká donekonečna
- `WaitForMultipleObjects ()`:
 - `count` \leq `MAXIMUM_WAIT_OBJECTS` (64)
 - `objects` – ukazateľ na pole objektov jadra
 - `waitAll` – čakať na prvý objekt, alebo na všetky?
 - Funkcia vráti index prvého signalizovaného objektu
- Vedľajšie efekty:
 - `Mutexy`, `auto-reset events` a `waitable timers` sú po ukončení `Wait*` funkcií opäť nesignalizované.

Otázky?

Ďakujem za pozornosť;)