

Sincronizzazione 8 MUTEX WINDOWS

martedì 25 novembre 2025 15:35

Analogamente a quanto detto per UNIX, andiamo a vedere in WINDOWS:

1. Creazione

Sono in pratica simili a semafori binari, ovvero dei semplici meccanismi per la mutua esclusione

```
HANDLE CreateMutex(LPSECURITY_ATTRIBUTES lpMutexAttributes,  
    BOOL bInitialOwner,  
    LPCTSTR lpName)
```

Descrizione

- invoca la creazione di un mutex

Restituzione

- handle al mutex in caso di successo, NULL in caso di fallimento

Parametri

- lpMutexAttributes: puntatore a una struttura SECURITY_ATTRIBUTES
- bInitialOwner: indicazione del processo chiamante come possessore del mutex
- lpName: nome del mutex

2. Apertura

```
HANDLE OpenMutex(DWORD dwDesiredAccess,  
    BOOL bInheritHandle,  
    LPCTSTR lpName)
```

Descrizione

- invoca l'apertura di un mutex

a.

- handle al mutex in caso di successo, NULL in caso di fallimento

Parametri

- dwDesiredAccess: accessi richiesti al mutex
- bInheritHandle: specifica se l'handle e' creditabile
- lpName: nome del mutex

3. Operazioni

Accesso al mutex

```
DWORD WaitForSingleObject(HANDLE hHandle,  
    DWORD dwMilliseconds)
```

Rilascio del mutex

```
BOOL ReleaseMutex(HANDLE hMutex)
```

a.

 0 in caso di fallimento

Un solo processo (o thread) viene eventualmente risvegliato per effetto del rilascio del mutex

b. Per il rilascio solo dal thread che lo acquisito in precedenza