WIKIPEDIA

Yield (multithreading)

In computer science, **yield** is an action that occurs in a computer program during multithreading, of forcing a processor to relinquish control of the current running thread, and sending it to the end of the running queue, of the same scheduling priority.

Contents

1 Examples

2 In coroutines

3 See also

4 References

Examples

Different programming languages implement yielding in various ways.

- `pthread_yield()` in the language C, a low level implementation, provided by POSIX Threads[1]
- `std::this_thread::yield()` in the language C++, introduced in C++11.
- The *Yield method* is provided in various object-oriented programming languages with multithreading support, such as C# and Java.[2] OOP languages generally provide class abstractions for thread objects.

In coroutines

Coroutines are a fine-grained concurrency primitive, which may be required to yield explicitly. They may enable specifying another function to take control. Coroutines that explicitly yield allow cooperative multitasking

See also

- Coroutines
- Java (software platform)
- Common Language Runtime
- Java virtual machine
- Actor model

References

1. "pthread_yield" (https://www.ibm.com/support/knowledgecenter/#!/SSLTBW_2.1.0/com.ibm.zos.v2r1.bpxbd00/ptyield.htm).
2. "Thread.yield" (http://www.javamex.com/tutorials/threads/yield.shtml). Javamex. Retrieved 24 June 2011.

Retrieved from "https://en.wikipedia.org/w/index.php?title=Yield_(multithreading)&oldid=772165168"

This page was last edited on 25 March 2017, at 18:12.