

ThreadOS C++ Port Continuation

By Nathan Pham ; Faculty Advisor: Erika Parsons

Introduction:

- ThreadOS - educational OS emulator
- Problems in Java ThreadOS:
- Limitations cause issues
- Poor documentation on the Java version

- Goals for C++ ThreadOS:
- Improved synchronization
- Better packaging methods
- Better documentation (in websites and code)

Purpose and Tasks:

- Solve a bug between Scheduler and Loader
- Create a new scheduling algorithm
- Verify the Filesystem classes
- Fix any bugs found along the way
- Remove and update deprecated documentation

Results:

- Switch Scheduler to Cooperative Multitasking Algorithm
- Ensure type-safety in Filesystem classes
- Correct behavior is displayed in all tests
- Removed deprecated information in HTML docs
- Added additional instructions in HTML docs

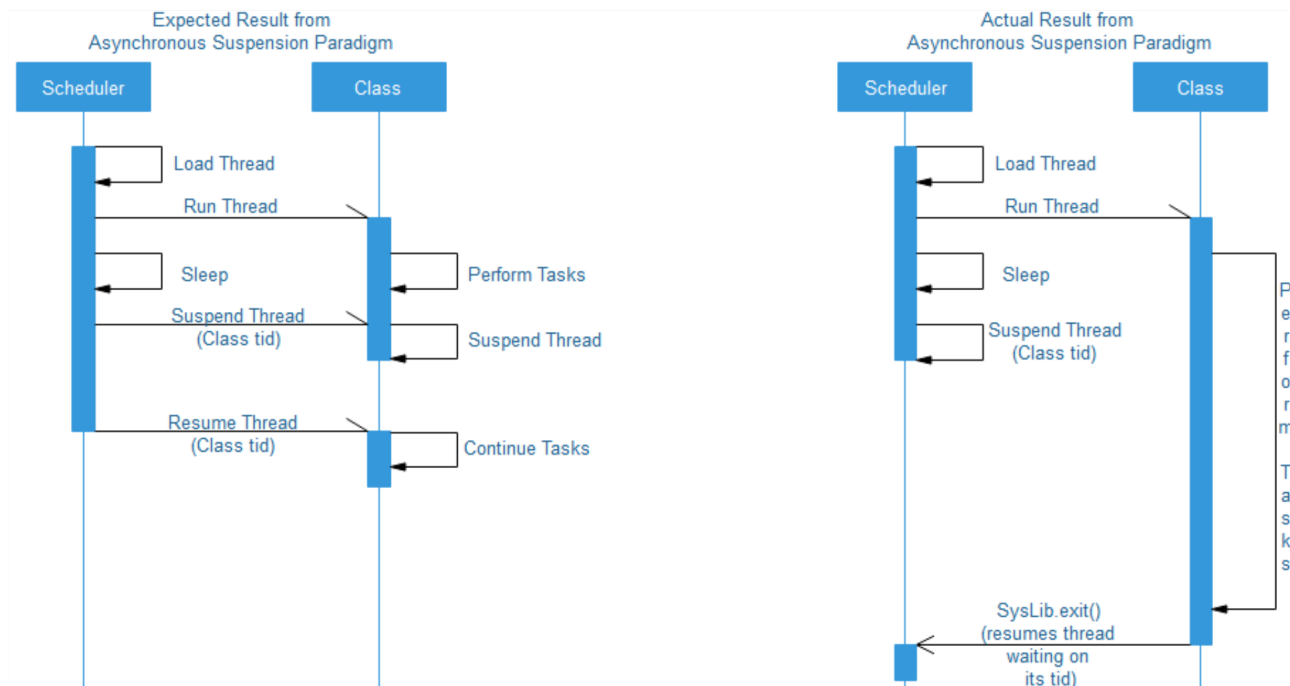


Figure 1: Scheduler Issue Illustration

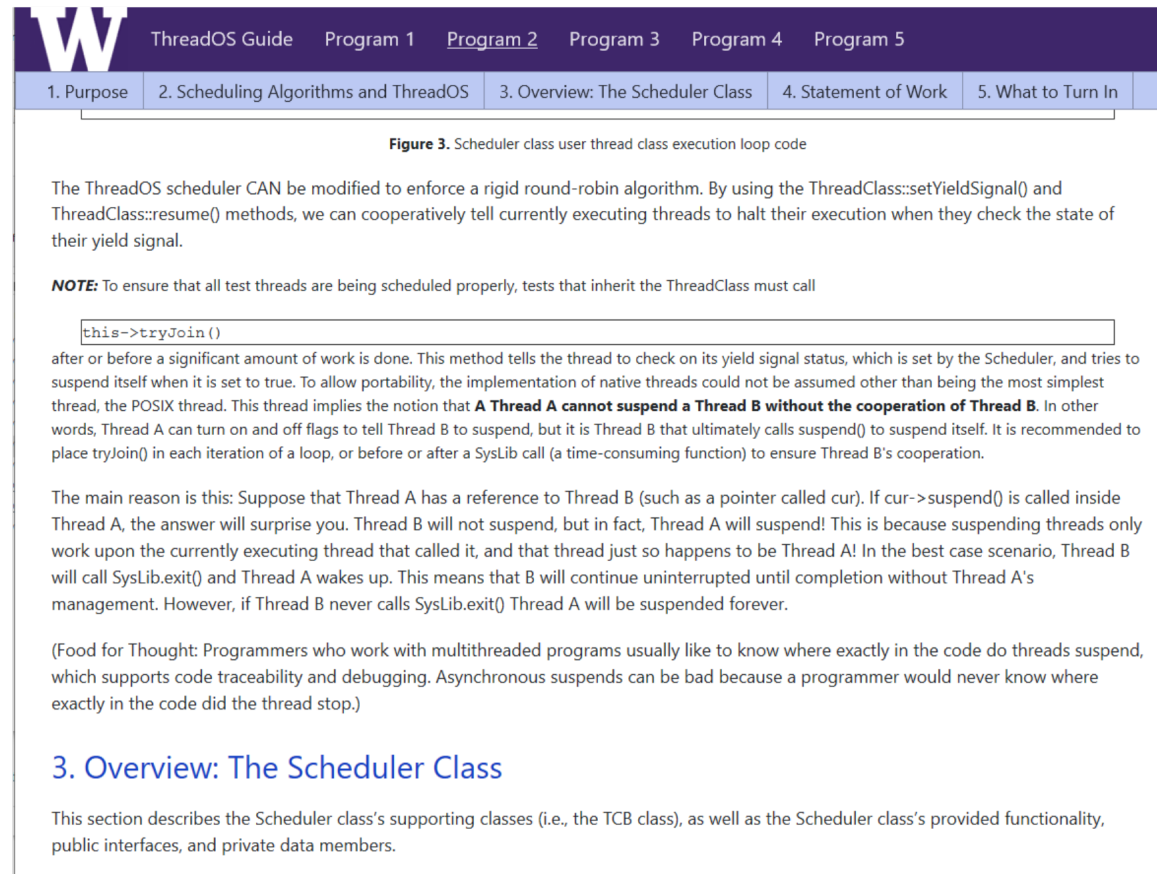


Figure 3. Website Modification Example

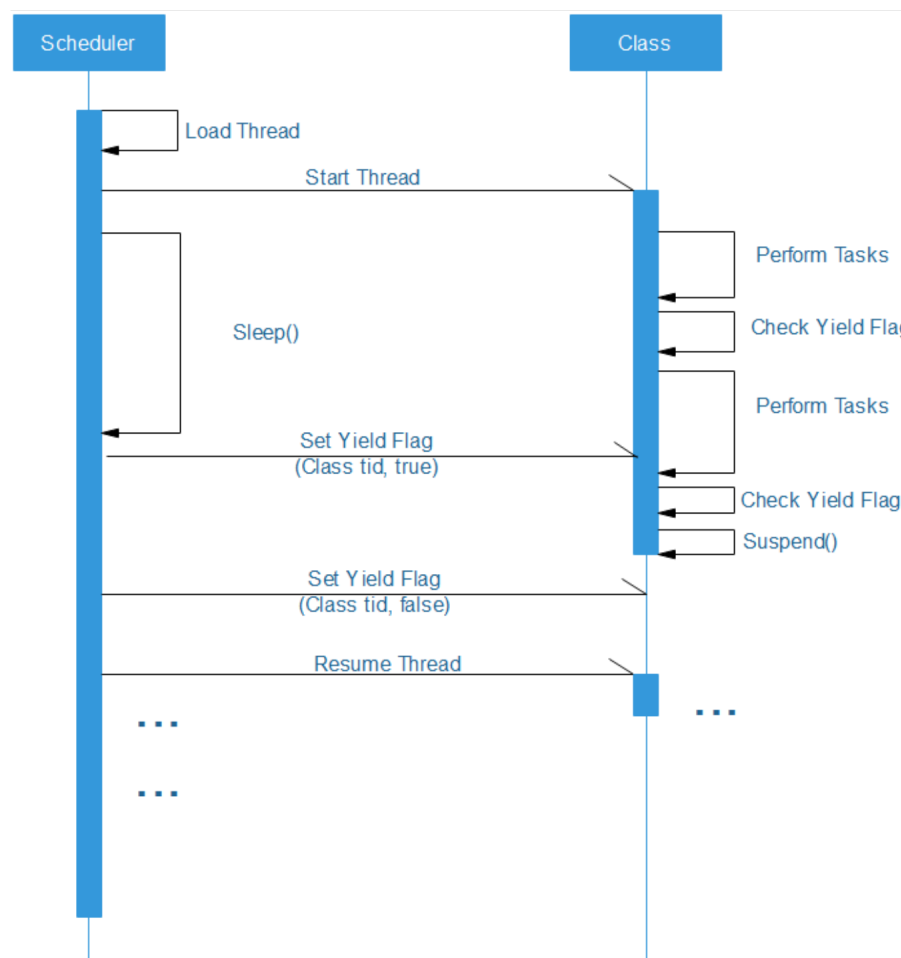


Figure 2. Cooperative Multitasking Example

What I learned:

- First experience contributing to a long-term project
- Linux and POSIX Thread functions and limitations
- Applying synchronization techniques
- Doxygen, GitLab, HTML
- OS emulation is quite fun!

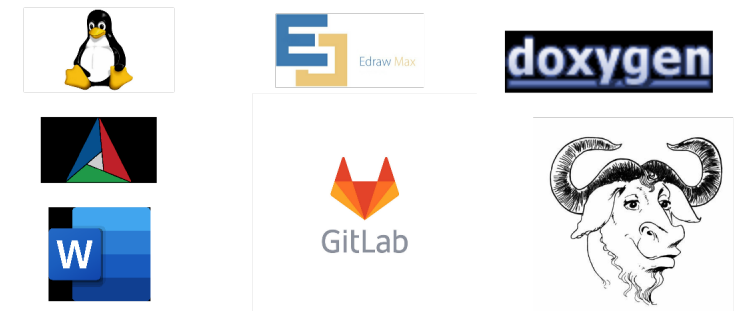
```
ThreadOS >> 1 Test5
sysExec( Test5 )
adding thread to scheduler
threadOS: a new thread (thread=Thread[Thread-2,2,main] tid=1 pid=0)
1: format( 48 ).....successfully completed
Correct behavior of format.....2
2: fd = open( "css430", "w+" )...successfully completed
Correct behavior of open.....2
3: size = write( fd, buf[16] )...successfully completed
Correct behavior of writing a few bytes.....2
4: close( fd ).....successfully completed
Correct behavior of close.....2
5: reopen and read from "css430"..successfully completed
Correct behavior of reading a few bytes.....2
6: append buf[32] to "css430"....successfully completed
Correct behavior of appending a few bytes.....1
7: seek and read from "css430"....successfully completed
Correct behavior of seeking in a small file.....1
8: open "css430" with w+.....successfully completed
Correct behavior of read/writing a small file.0.5
9: fd = open( "bothell", "w" )...successfully completed
10: size = write( fd, buf[6656] ).successfully completed
Correct behavior of writing a lot of bytes....0.5
11: close( fd ).....successfully completed
12: reopen and read from "bothell"successfully completed
Correct behavior of reading a lot of bytes....0.5
13: append buf[32] to "bothell"...successfully completed
Correct behavior of appending to a large file.0.5
14: seek and read from "bothell"...successfully completed
Correct behavior of seeking in a large file...0.5
15: open "bothell" with w+.....successfully completed
Correct behavior of read/writing a large file.0.5
16: delete("css430").....successfully completed
Correct behavior of delete.....0.5
17: create uwb0-29 of 512*13.....successfully completed
Correct behavior of creating over 40 files ...0.5
18: uwb0 read b/w Test5 & Test6...
```

Figure 4. Filesystem Test Validation

Future Work:

- Cross-Platform Packaging Support
- Automated Testing (optional)

Tools Used:



Special Thanks to Erika Parsons and Ardan Ahanchi