

発表概要

マルチスレッドを学習するためのメモリアクセス可視化ツールの提案および実装

吉塚 大浩^{1,a)} 篠埜 功^{1,b)}

2021年1月14日発表

マルチスレッドプログラムでは、スレッド間で共有しているデータにアクセスする際、他スレッドの処理待ちや排他制御を行うことで、データにアクセスする順番を制御している。そのため、スレッドの実行順序の制御を把握することは、マルチスレッドプログラムを学習するうえで必要不可欠である。これをマルチスレッドの初学者が自習する場合、教科書などに載っているサンプルコードを見ながら自身でプログラムを書くことがある。しかし、サンプルプログラムを打ち込んで実行するだけでは、実行順序の制御を把握するのは難しい。この点は、様々なマルチスレッドプログラムにおいて、実行中のソースコードの場所とともに変数情報などを視覚化することで補うことができると考えた。この考えに基づき、本発表では、マルチスレッドの学習を支援することを目的としたメモリアクセス可視化ツールを提案する。このツールは、ステップ実行を行いながら、スレッドごとに実行中のソースコードの場所に色をつける。また変数情報を表示する機能があり、どのスレッドが各変数を更新したかを色で表す。このツールは、C言語を対象にLLDBのAPIを使って実装した。また、学習者が教科書とサンプルコードを用いて自習した場合と本ツールを用いて自習した場合の2つに分け、テストとアンケートを用いて、スレッドの実行順序の制御について理解が深まったかどうか評価を行う。

Presentation Abstract

A Tool for Learning Multithread Programming by Memory Access Visualization

TAKAHIRO YOSHIKAZU^{1,a)} ISAO SASANO^{1,b)}

Presented: January 14, 2021

In multithreaded programs, when accessing data shared among threads, the order in which the data is accessed is controlled by waiting for other threads having finished or by mutual exclusion. Therefore, understanding the control of the order of thread execution is essential for learning multithreaded programs. When beginners of multithreaded programming study by themselves, they may write programs while looking at some sample code in textbooks and other sources. However, it is difficult to grasp the control of the execution order just by typing and executing the sample code. We consider that this point can be overcome by visualizing variable information as well as the executing part in the source code while executing various multithreaded programs. Based on this idea, in this presentation, we propose a memory access visualization tool that aims to support beginners to learn multithreaded programming. The tool has a functionality for step execution with the executing part in the source code being colored for each thread. Also, the tool has a functionality for showing variable information with a color indicating which thread updated which variable. The tool was implemented for C using LLDB API. To evaluate the tool, we asked some beginners of multithreaded programming to use the tool. We divide them into two groups, one with the textbook and sample code and the other with the tool. After learning, we do tests and questionnaires in both groups to evaluate to what extent the beginners have understood the control of the order of thread execution.

This is the abstract of an unrefereed presentation, and it should not preclude subsequent publication.

¹ 芝浦工業大学
Shibaura Institute of Technology, Koto, Tokyo 135-8548,
Japan

a) ma19092@shibaura-it.ac.jp

b) sasano@sic.shibaura-it.ac.jp