

BYOD 下におけるプログラミング演習環境の統一に向けて

玉田 春昭^{†1}

概要: プログラミング演習において、受講生間での演習環境の統一は重要である。頻繁なバージョンアップが行われている昨今では特に重要な課題である。そこで本稿では、Visual Studio Code で利用できる Remote Containers プラグインを用いた環境統一手法（以降 devcontainer）について述べる。そして、devcontainer で構築したい環境や、利点、欠点について議論する。

1. はじめに

多くの高等教育機関において、BYOD (Bring Your Own Device) が浸透して久しい [1]。プログラミング教育では、プログラムの実行環境の統一が望ましいが、BYOD の場合完全な統一は難しい。昨今では、プログラミング言語自体のバージョンアップも頻繁に行われている。バージョンが更新されると新たな API の登場や古い API が非推奨に指定されることがある。これにより、過去の講義資料が誤りになる可能性がある。これを防ぐためには、バージョンアップによる更新内容と、講義資料の内容の両方を詳細に把握する必要がある。しかしながら、これには非常に多大なコストが必要となる。

特に Java 言語は半年ごとに新しいバージョンがリリースされている*1。この場合、複数のバージョンが利用可能であるため、指定したバージョン以外を利用する受講生も存在する可能性がある。加えて、再履修生の場合、古い環境を更新しないまま授業に臨む場合もある。

これを解決するため、オンラインのプログラミング環境を用意し、演習に利用する方法が古くから提案されている [2]。ただし、このような環境の準備に多大なコストを要する点が問題となる。一方で近年では、Google Colaboratory*2 や Replit*3, GitHub Codespaces*4, CodeRunner*5などのオンライン IDE も充実している。ただしこれらの場合、教育用のために用意した外部ツールを導入するのに手間がかかるのが問題である。

これらを解決するため、本稿では Visual Studio Code を用いた devcontainer*6 による演習環境の構築と、利点、欠点について述べる。devcontainer とは、Docker コンテナにローカル環境のソースコードをマウントし、コンテナ内の環境で開発を進める。そして、そのコンテナ内のファイルをローカル環境の Visual Studio Code から編集するというものである (図 1)。なお、devcontainer の導入には、コンテナイメージ構築のための Dockerfile の作成と devcontainer 環境構築の設定ファイル群の準備が必要である。

2. これまでの演習環境統一への試み

これまでに演習環境を統一するために、様々な試みが行われてきた。それらの手法と問題点を表 1 に示す。

現状の著者の勤務校では、受講生の環境を統一するため、Homebrew*7 や asdf*8, Sdkman*9 などのバージョン管理システムを用いており、(2) に相当する。

3. 演習環境統一で行いたいこと

演習環境の統一以外にも以下のことが期待できる。

品質測定 プログラミング演習で提出される課題プログラムの評価観点は大きく、正しさ、品質、その他の 3 つ

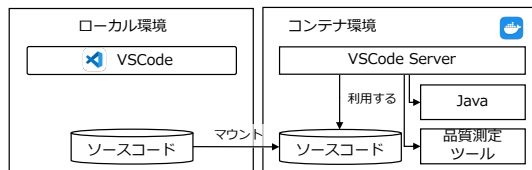


図 1 devcontainer の構成図

^{†1} 現在、京都産業大学

Presently with Kyoto Sangyo University

*1 <https://www.oracle.com/java/technologies/java-se-support-roadmap.html>

*2 <https://colab.research.google.com/>

*3 <https://replit.com/>

*4 <https://github.com/features/codespaces>

*5 <https://coderunner.org.nz/>

*6 <https://code.visualstudio.com/docs/remote/containers>

*7 <https://brew.sh/>

*8 <https://asdf-vm.com/>

*9 <https://sdkman.io/>

表 1 演習環境統一手法とそれらの問題点

方法	概要	問題点
(0) 情報処理教室	学内の教室に PC を設置し、必要なツール群をインストールしておく。各受講生はその PC を利用する。	運用に多大なコストが必要となる。特定ツールの導入は難しい。BYOD 化により採用できない場合が多い。
(1) 各受講生による個別インストール	各受講生が講義資料に書かれた指示に従って、個別に必要なソフトウェアをインストールする。	インストールサポートが必要。再履修生は古い環境のままである場合がある。
(2) インストール作業の自動化	シェルスクリプトなどで必要なソフトウェアをインストールする。	再履修生は古い環境のままである場合がある。インストールスクリプトのメンテナンスコストが必要になる。
(3) ネットブート [3]	ネットワーク経由で OS イメージをダウンロードし、起動する。PC は thin クライアントとして用いる。	環境構築、運用に大きなコストを要する。受講生のいつもの環境とは異なる環境で演習に臨むことになる。
(4) 仮想マシン	必要なソフトウェアがインストールされた仮想マシンイメージを配布し、その仮想マシン上で演習を実施する。	受講生のいつもの環境とは異なる環境で演習に臨むことになる。必要とするディスク容量が大きい。OS のライセンスに注意が必要である。
(5) オンラインエディタ	ブラウザから扱えるエディタを用いて演習を行う。演習向けに環境構築が必要なケース、一般向けのオンラインエディタを利用するケースがある。	環境構築が必要なケースでは運営に大きなコストを要する。オンラインエディタを利用するケースでは、演習用の独自ツールの導入が難しい場合がある。
(6) devcontainer	Docker コンテナ上にマウントされたソースコードをローカル環境の Visual Studio Code から編集する。	エディタが Visual Studio Code に限定される。シェル環境が受講生の環境とは異なる可能性がある。

に分類できる。受講生自らが自身のプログラムの品質を測定できる環境を用意する。例えば華山らが提案するように、品質の可視化による教育手法の導入も期待できる [4]。

自動採点 課題の出題形式は、大きく分けて、写経、穴埋め、デバッグ、文章題、自由課題の 5 つに分類できる。文章題では、文章で与えられた仕様に基づいてプログラムを作成する。このような問題が正しく実装できたかを受講生自身が評価するため、単体テストの活用により、プログラムの自動評価を狙う。

進捗把握 演習中に課題の進捗状況をリアルタイムに把握することは、きめ細やかな指導のためには不可欠である。従来から様々な手法で取り組まれている [5]。

4. devcontainer の利点と欠点

- 利点
 - ツールの同梱が容易。著名なツール、独自のツールのどちらもほぼ同じ手順で同梱できる。
 - 受講生のローカル環境を汚さない。
 - 受講生の利用 OS に依存しない。
- 欠点
 - Visual Studio Code に限定される。
 - 実行環境の理解が、受講生にはハードルが高い。
 - ターミナルでの操作も Visual Studio Code 上に限定される。ただし、devcontainer cli が 2022 年 5 月より提供されている^{*10}。しかし、利用するには実行環境の理解がある程度必要となるため、初学者向けの講義で利用することは困難であると思われる。

5. まとめと議論の種

本稿では、プログラミング演習科目において、受講生の

プログラミング環境を統一するための準備、運営が簡便な方法である devcontainer について述べた。実際に導入した事例を見ると、devcontainer は必ずしも全受講生に受け入れられるわけではない [6]。また、devcontainer のセットアップが難しい受講生が 18%いたと報告されている。devcontainer への議論の種として、以下の点を挙げる。

- 利用する是非、
- devcontainer が活きる言語、
- 導入したいツール等、
- 運用ノウハウ、
- 受講生に求める実行環境への理解の度合い。

謝辞

この研究は JSPS 科研費 20K11761 の助成を受けた。

参考文献

- [1] ICT 利活用調査部会：2016 年度 BYOD を活用した教育改善に関する調査研究概要と報告，技術報告，大学 ICT 推進協議会 (2018)。
- [2] Goldman, M., Little, G. and Miller, R. C.: Collabode: Collaborative Coding in the Browser, *Proc. 4th International Workshop on Cooperative and Human Aspects of Software Engineering*, p. 6568 (2011).
- [3] 大本英徹：NetBoot による学生ノートパソコンの教育専用端末化環境の構築と運用，情報処理学会関西支部大会 (2009)。
- [4] 華山魁生，まつ本真佑，肥後芳樹，楠本真二：プログラミング教育における実績可視化システムの提案と評価，情報処理学会論文誌，Vol. 61, No. 3, pp. 644-656 (2020)。
- [5] 榎原絵里奈，井垣 宏，吉田則裕，藤原賢二，飯田 元：プログラミング演習における探索的プログラミング行動の自動検出手法の提案，コンピュータソフトウェア，Vol. 35, No. 1 (2018)。
- [6] Valstar, S., Griswold, W. G. and Porter, L.: Using DevContainers to Standardize Student Development Environments: An Experience Report, *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '20, pp. 377-383 (2020)。

^{*10} <https://code.visualstudio.com/blogs/2022/05/18/dev-container-cli>