

Web コンソールを利用したプログラミング実習環境の活用

坂田 圭司

東海大学

kgsakata@tsc.u-tokai.ac.jp

今後のプログラミング学習において、人工知能の活用を含めた多様性に対応するために、多くのプログラミング言語やライブラリを利用可能な学習環境を整備する必要がある。プログラミングコンテスト等や大学の授業で一般的なC/C++言語に代表されるコンソール開発環境は、インストールが必要な場合が多く、コンピュータ室への導入ではセキュリティ設定やバージョンアップの難しさが問題になってきた。また、Visual StudioやEclipse等の統合開発環境（IDE）は高機能すぎて、学習内容に注力しにくい問題がある。本研究では、上記の問題点を解決するために、Web 開発環境である Google Colaboratory のコンソール機能を用いて、プログラミングの教材提示および実習を行う具体的な方法について提案を行う。

1. 研究の背景と目的

新学習指導要領によってプログラミング学習の重要性が増す中、人工知能の活用を含めた多様性に対応するために、多くのプログラミング言語やライブラリを利用可能な学習環境を整備する必要がある。

2022年度現在において、ビジュアルプログラミング言語やPythonにおいてWeb実習環境が充実して、コンピュータ室や各自所有のPCにおいても実習しやすいため授業での採用が多くなっている。しかし、プログラミングコンテスト等や大学の授業で一般的なC/C++言語に代表されるコンソール開発環境は、インストールが必要な場合が多く、コンピュータ室への導入ではセキュリティ設定やバージョンアップの難しさが問題になってきた。

またVisual StudioやEclipse等の統合開発環境（IDE）は高機能すぎて、学習内容に注力しにくい問題がある。プログラミングの理解度が低い学生は、開発環境の導入や操作に躓く傾向があり、より学習進度を遅くする影響が出る。

本研究では、上記の問題点を解決するために、Web 開発環境である Google Colaboratory のコンソール機能を用いたプログラミングの教材提示および実習方法を提案する。この方法によって、開発環境の準備や操作を最低限に抑え、学習への注力向上を期待できる。

2. Google Colaboratory におけるシステムの構成

Google Colaboratory は、Google が無料で提供している Web 開発環境で、操作体系およびファイル保存形式は Python の学習環境として広く使わ

れている Jupyter Notebook と互換性を持つ。

利用者が Google Colaboratory にアクセスすると仮想環境が起動して、この仮想環境上で Jupyter Notebook 互換の Python 開発環境が動作する（図1）。開発環境においてコードやテキストを記述するノートブックは、Google ドライブ上の拡張子 ipynb ファイルにマウントされ、利用者がノートブックを更新すると自動保存される。この自動保存機能によって利用者のコード保存し忘れが無いため、実習時のトラブルを防ぐ利点となる。

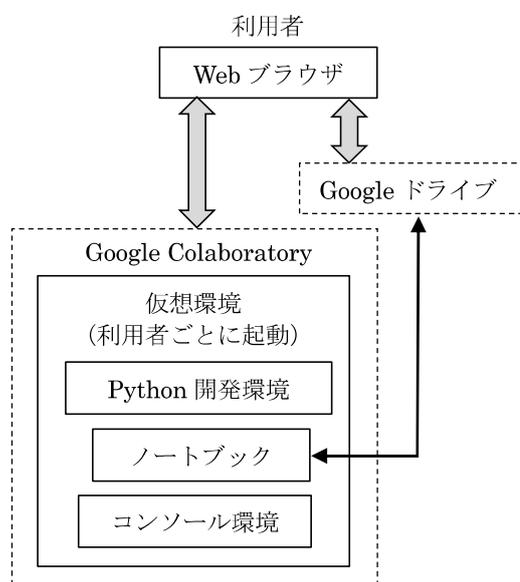


図1 Google Colaboratory の動作環境

仮想環境には Linux コンソール環境が含まれている。この環境には表1に示す各種コンパイラが導入済みのため、Python 開発環境との連携して、複数のプログラミング言語による実習が可能になる。

表 1 導入済みの主なコンパイラ

コンパイラ	機能
clang / clang++	C/C++コンパイラ
gcc / g++	C/C++コンパイラ
javac	Java 開発環境 (JDK)

3. Google Colaboratory における Web コンソールの利用方法

ノートブック上のコード先頭に!を付けて実行すると、コンソールコマンドとして解釈され実行される。実行結果の標準出力はコードの下部に表示される。



図 2 コンソールコマンドの呼び出しと出力

ソースコードとの連携は、仮想環境のストレージにソースファイルを配置する方法が操作がシンプルで使いやすい。コード先頭に「%%file ソースファイル名」を記述して実行すると、次行以降の内容が、仮想環境上のディレクトリにファイル保存される。

例として C++言語プログラムの実行手順を記述する (図 3)。

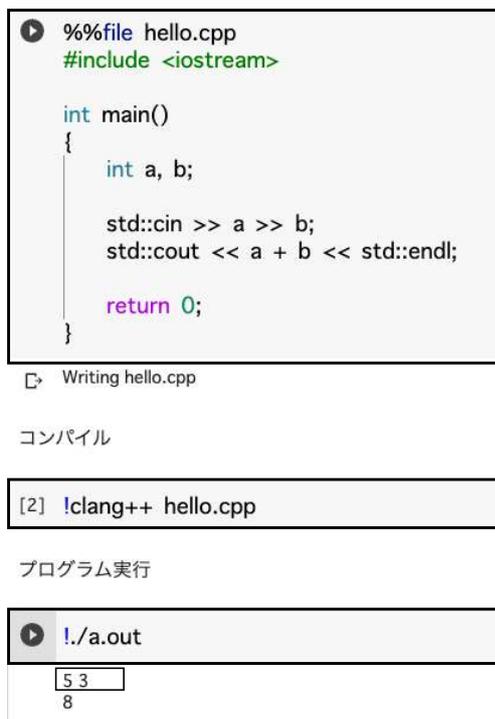


図 3 C++言語における実行までの手順

4. 本利用方法によって期待できる効果

情報科教育において、従来のコンソールアプリケーションのプログラミング環境は、教員側としては実行環境の整備と保守に手間がかかり、授業運営においては生徒の操作習熟に時間がかかるため、授業での実施は少ない。

しかし、組込みシステム開発やプログラミングコンテストに対応した本格的なスキルを修得するためには、コンソールにおけるプログラミング経験は欠かせない要素である。

今回提示した方法では、Jupyter Notebook をベースとしたテキストとコードを混在させた学習環境を利用できるため、プログラミング内容と実行手順への集中した学習が可能になる。

操作上の長所としては、標準入力テキストフォーム入力と接続されていることで、直感的な利用がしやすくなっている点が挙げられる (図 3 のプログラム実行部分)。従来の GUI 環境上に構築されたコンソールの標準入力は、ダイアログやタブなどを用いて特殊なキー入力を強いる問題があった。

実行環境の整備と保守については、生徒毎の Gmail アドレス取得が前提となる。この問題を解決して Google ドライブが利用可能な状況であれば、Google Colaboratory は Web ブラウザ上で動作するため、PC だけでなくタブレット端末または Chromebook でもコンソールを用いたプログラミングが可能であるため、短時間での体験的なプログラミングから本格的な実習までの利用が期待できる。

5. 参考資料

Colaboratory へようこそ (公式チュートリアル)
<https://colab.research.google.com/>