

第1章

Visual Studio CodeをちょっとしたIDEに!
定番「Hello World」の実行までPCでサッと試せる
Rust開発環境の構築

特別付録として、本誌2020年5月号特集「C/C++後継 モダン言語の研究」の65～76ページを再掲載しました。
 ビギナー向けに、手元PC (Windows/Linux/macOS) で試してもらうことを目的としています。

中林 智之

コンパイラ&ツールチェーンを入れる

● バージョン

Rustをマイコンで動かす前に、まずは、ホスト環境向けにRustの環境を構築しましょう(図1)。筆者の環境はUbuntu 22.04です。実験で利用するRustのバージョンは1.67.1です。

● インストール

Rustのインストールは非常に簡単です。Rustのインストーラであるrustupを使用します。次の手順に従って、rustup(<https://www.rust-lang.org/tools/install>)をインストールします。LinuxやmacOS, その他のUNIXライクなOSでは、ターミナルで次のコマンドを実行するだけでインストールが完了します。

```
$ curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

インストール方法を尋ねられますが、デフォルト(default)でOKです。

```
# snip
```

```
1) Proceed with installation(default)
2) Customize installation
3) Cancel installation
>1
```

```
# snip
```

次の1文が表示されればインストール完了です。

```
Rust is installed now. Great!
```

次のコマンドでインストールが完了したかどうか、確認できます。rustcが見つからない場合は、環境変数PATHの設定を確認してください。

```
$ rustc -V
```

```
rustc 1.67.1 (d5a82bbd2 2023-02-07)
```

Windowsの場合、公式サイトにアクセスすると専用ページが表示されます。インストーラをダウンロード

し、デフォルトの設定でインストールすれば、必要なツール一式が使用できます。

ホスト環境をターゲットとしたビルドでは、リンカが別に必要です。Linux/macOS環境ではgccを、Windows環境ではVisual C++ Build Toolsをインストールしてください。

注目エディタVSCodeによるIDE化

筆者が普段Rustの開発で使用しているのはVisual Studio Code(以下VSCode)とRust(rls) Extensionです。Rust(rls) Extensionを追加することで、コンパイル・エラー表示、コード補完、関数や構造体定義へのジャンプ、ドキュメント表示、ユニット・テストの個別実行が行えるようになり、ちょっとしたIDEとして使うことができます。

● 本体のインストール

VSCodeのウェブ・ページにアクセス(<https://code.visualstudio.com/>)します。OSごとに

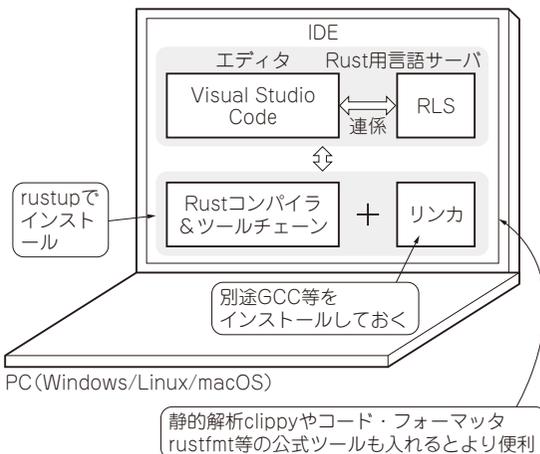


図1 注目のエディタVisual Studio CodeによるRust開発環境の全体像