

Geant4-V11 を Windows(WSL2)にインストールする手順書

(注)WSL2 は、Windows10 バージョン 2004 以降(ビルド 19401 以降)または Windows11 で利用可能な Linux システムです。

(概要)

この手順書では、Windows の WSL2 に Geant4 version11 をインストールする手順を説明します。Geant4 に必要なコンパイラ等の環境は、WSL2 上の Ubuntu に導入する方法を取ります。また Windows のアプリも 2 つ導入します。

(作業の流れ) * Geant4 のバージョンアップは、(3)(4)のみ作業してください。

- 1) Windows に WSL2 を導入して Ubuntu をインストールする
- 2) Geant4 に必要な開発環境のセットアップ
- 3) Geant4 のインストール
- 4) Geant4 サンプルコードの実行確認
- 5) CERN ROOT のインストール
- 6) Windows のアプリインストールする (オプションです。スキップ可。)

(設定手順)

1) Windows に WSL2 を導入して Ubuntu をインストールする

ここでは、その手順を簡潔に記載します。Windows10 バージョン 2004 以降(19041 以降)または Windows11 が対象です。このマニュアルは Windows11 にて、動作確認しています。

詳細な手順は、マイクロソフトの Web サイトで確認することができます。

Microsoft 「WSL を使用して Windows に Linux をインストールする」

<https://docs.microsoft.com/ja-jp/windows/wsl/install>

以下に手順を示します。

- ・ Windows アップデートを行なっておいてください。
- ・ PowerShell またはコマンドプロンプトを“管理者として実行”を選んで開きます。

以下、PowerShell を使うことにします。

* 以下の説明では、Windows のプロンプトは”>”で表し、Ubuntu のプロンプトは”\$”で表します。

* 以下の説明に沿って進めない場合は、資料末尾の補足も参考にしてください。

- ・ PowerShell で次のコマンドを入力します。

```
> wsl --install
```

- ・デフォルトで WSL2 と Ubuntu がインストールされます。
- ・Windows システムを再起動するメッセージが表示されます。Windows を再起動します。
- ・再起動後に WSL2 の PowerShell ウィンドウで「**ユーザ名**」と「**パスワード**」の**設定が求められます**。これが **Ubuntu 上の管理者アカウント**になります。(英数字であれば Windows と同じアカウントでも構いません。)(WSL2 のウィンドウが開いていない場合は、PowerShell を開いて、wsl とコマンド入力してください)
- ・これで WSL と Ubuntu のインストール完了です。PowerShell の画面では、プロンプトが **ユーザ名@コンピュータ名**になっているはずです。これで Ubuntu にログインした状態になっていることが確認できます。
- ・WSL2 の Ubuntu システムからログアウトするには、exit と入力します。
- ・但し、exit コマンドでは、ユーザがログアウトしただけで、バックグラウンドで Ubuntu は実行状態になっています。
- ・確認のためバージョン確認などのコマンドを試しましょう。この確認は Windows で PowerShell を開いて、次のコマンドを実行します。

WSL の動作確認

```
> wsl -l -v
```

(表示内容)

| NAME | STATE | VERSION |
|---------|---------|---------|
| *Ubuntu | Running | 2 |

NAME, STATE, VERSION が表示されます。Ubuntu が Running になっており、(WSL の)VERSION は 2 であることが確認できます。

- ・WSL を終了するには、Windows の PowerShell で次のように実行します。

WSL の全てを終了する

```
> wsl --shutdown
```

- ・先ほどのコマンドで Ubuntu が Stopped になっていることを確認しましょう。

```
wsl -l -v
```

| NAME | STATE | VERSION |
|---------|---------|---------|
| *Ubuntu | Stopped | 2 |

(Tips) WSL の Ubuntu と Windows 間でのファイルのやりとりについて

WSL2 から、お互いのファイルシステムにアクセスできるようになりました。

・ Windows から Linux のファイルにアクセス

エクスプローラのファイルパスの欄に次のように入力することでアクセスできます。

[\\wsl\\$](#)

インストールしてある LinuxOS がフォルダとして表示されます。

・ Linux から Windows のファイルにアクセス

Windows の C ドライブは、次の場所にマウントされています。

/mnt/c

2) Geant4 に必要な開発環境のセットアップ

WSL の Ubuntu を起動します。

- ・ PowerShell を開いて、次のコマンドで起動します。

```
> wsl ~
```

- * 起動する場合は、いつもこのコマンドで行いましょう。ログイン後のプロンプトは、`user@host:~$` のようになります。user と host は、ユーザ名とコンピュータ名です。
もしも起動後のプロンプトが、`user@host:/mnt/c/windows/system32$` となっている場合は、必ず初めに `$ cd ~` と入力して、プロンプトが `user@host:~$` となるようにしましょう。

- ・ Ubuntu の更新リストをアップデートします。この際に Ubuntu の管理者パスワードの入力が求められます。WSL でのインストール時に設定したものを入力します。

```
$ sudo apt update
```

```
$ sudo apt upgrade
```

```
...
```

```
Do you want to continue [Y/n] Y と入力して続けてください。
```

- ・ X11 のアプリを追加して、使用可能かを確かめます。この時も Ubuntu の管理者パスワード入力が求められたら入力してください。

```
$ sudo apt install x11-apps
```

- ・ xeyes というサンプルを実行してみます。

```
$ xeyes
```

目のウィンドウが表示されて、マウスポインタの方を見ていれば動作確認 OK です。

- ・ PowerShell のウィンドウをアクティブにして、Ctrl キーを押しながら c キーを押して終了（実行をキャンセル）します。

- ・必要なライブラリやコンパイラなどをインストールします。

次のコマンドを順番に実行してインストールしていきます。それぞれインストールを続けるかの確認[Y/n]の入力が求められます。また、エラー(E)が出ることがあります。エラーの場合には、もう一度、同じコマンドを実行して見てください。

```
$ sudo apt install build-essential
$ sudo apt install cmake
$ sudo apt install expat
$ sudo apt install libexpat1-dev
$ sudo apt install xorg-dev
$ sudo apt install freeglut3-dev
$ sudo apt install qtbase5-dev
$ sudo apt install qtbase5-dev-tools
$ sudo apt install libxerces-c-dev
$ sudo apt install emacs-nox
$ sudo apt install qt3d5-dev
$ sudo apt install qt3d5-dev-tools
$ sudo apt install python3          (オプション)
$ sudo apt install python3-dev      (オプション)
$ sudo apt install openssl
```

- ・Ubuntu やコンパイラのバージョンを確認しておきましょう。

```
$ cat /etc/lsb-release
(多分、OS のバージョンは Ubuntu 22.04.5 LTS です。)
$ gcc --version
$ g++ --version
(多分、共にコンパイラのバージョンは 11.4.0 です。)
```

3) Geant4 のインストール

- ・ソースコードをダウンロードしてします。Windows のブラウザでダウンロードしましょう。([geant4-v11.3.0.tar.gz](https://geant4.org/download))

ダウンロード: <https://geant4.org/download>

「Source code」の「Download tar.gz」のボタンをクリックしてダウンロードします。

- ・ファイルを展開します。今回は、Ubuntu のホームディレクトリに geant4 というフォルダを作成して、その中に展開します。

(*){user}の部分は自分の windows のユーザ名です。例) /mnt/c/Users/admin/Downloads

```
$ cd
$ mkdir geant4
$ cd geant4
$ tar zxvf /mnt/c/Users/{user}/Downloads/geant4-v11.3.0.tar.gz
```

- ・cmake を実行します。下記のコマンドで¥マークがついている部分は、¥マークや改行を入れずに1行で続けて入力してくださいという意味です。

```
$ pwd
/home/{user}/geant4 と表示されることを確認してください。
$ mkdir geant4-v11.3.0-build
$ mkdir geant4-v11.3.0-install
$ cd geant4-v11.3.0-build
$ cmake -DCMAKE_INSTALL_PREFIX=../geant4-v11.3.0-install ¥
        -DGEANT4_INSTALL_DATA=ON ¥
        -DGEANT4_USE_OPENGL_X11=ON ¥
        -DGEANT4_USE_QT=ON ¥
        -DGEANT4_USE_GDML=ON ¥
        ../geant4-v11.3.0
```

*cmake を実行し終えた時に、エラーや Not found の様な出力表示が出ていないか確認してください。もしもエラーが出ていたら、コマンドの打ち間違いの可能性が高いです。その場合には、フォルダ内のファイルを消したのちに cmake を再実行してください。

```
$ pwd
/home/{user}/geant4/geant4-v11.3.0-build であることを確認する。
$ rm -rf * (ファイルを削除)
$ cmake -DCMAKE_INSTALL_PREFIX=.... (再確認して再実行)
```

- ・ビルドします。

```
$ make -j 4 (4 はスレッド数なので変更可)
エラー等がないかは確認してください。
$ make install
```

* make -j で指定するスレッド数は、コンピュータの性能で決めてください。但し、余り大きい数を指定した場合は、ファイルの順番が崩れるのかエラーになる場合がある様です。その場合には、スレッド数を減らして再度、make -j を実行してください。

- ・ログイン時の設定ファイルに環境変数を設定します。emacs エディタで編集します。

```
$ emacs ~/.bashrc
```

- ・開いたファイルの末尾に次の行を追加して保存します。

```
. /home/{user}/geant4/geant4-v11.3.0-install/bin/geant4.sh
```

Ctrl キーを押しながら、s キーを押して保存します。その後に、Ctrl キーを押しながら、x キー、c キーを続けて押して emacs エディタを終了します。

- ・一度ログアウトします。

```
$ exit
```

- ・ログインし直します

```
> wsl ~
```

*ログイン時にエラーが表示されないかを確認します。エラーがあれば、.bashrc の編集ミスと思われます。綴り間違いや、geant4 のパスに間違いがある可能性が高いのでエディタで確認して修正してください。

- ・設定された環境変数を確認します。

```
$ printenv | grep -i Geant4  
GEANT4_DATA_DIR=/home/...
```

*Geant4 のデータファイルのパスが出力されていれば OK です。

4) Geant4 サンプルコードの実行確認

サンプルコードをビルドして実行できることを確認します。ホームディレクトリに work というフォルダを作って、その中で作業を行います。

```
$ cd
$ mkdir work
$ cd work
$ cp -r /home/{user}/geant4/geant4-v11.3.0/examples/basic/B4/B4c .
$ mkdir B4c-build
$ cd B4c-build
$ cmake ../B4c
$ make -j 4
$ ./exampleB4c
```

・QT ウィンドウの「Session:」に次の入力を行って飛跡を確認してみます。

```
/run/beamOn 10
(飛跡が表示されるか確認しましょう)
exit (終了します)
```

・スレッドモードでのマクロファイル実行も試してみましょう。-t で4 スレッドを指定して、マクロファイル run2.mac を実行します。

```
$ ./exampleB4c -t 4 -m run2.mac
```

画面表示の中に、G4WT0～G4WT3 までの表示があり、4 スレッドで実行していると確認できます。

・実行後の出力ファイル名を確認します。

```
$ ls -l
B4.root というファイルが、実行後の時間にできたことが確認できます。
```

5) CERN ROOT のインストール

Geant4 のサンプルプログラムで出力された B4.root は、CERN の ROOT という解析ツールの出力形式となっています。CERN ROOT もインストールします。

ダウンロード URL <https://root.cern.ch/releases/release-63004/>

「Source distribution」にあるソースファイルを Edge でダウンロードしましょう。

([root-v6.30.04.source.tar.gz](https://root.cern.ch/releases/release-63004/root-v6.30.04.source.tar.gz))

- ・ホームディレクトリに root というフォルダを作成して、その中にダウンロードファイルを展開します。({user}部分は、自分のユーザ名です。)

```
$ cd
$ mkdir root
$ cd root
$ tar xvf /mnt/c/Users/{user}/Downloads/root-v6.30.04.source.tar
(または $ tar zxvf /mnt/c/Users/{user}/Downloads/root-v6.30.04.source.tar.gz )
$ ls (展開してできたフォルダを確認します。)
```

root-6.30.04 という名前のフォルダができます。(展開すると v がないみたいです。)

- ・ビルドします。

```
$ pwd
/home/{user}/root/ (一応、現在のフォルダを確認)
$ mkdir root-6.30.04-build
$ cd root-6.30.04-build
$ cmake -Dxrootd=OFF -Dbuildin_xrootd=OFF ../root-6.30.04
$ make -j 4
```

- ・実行の確認

```
$ . ./bin/thisroot.sh (注意！ピリオドから始まっています。)
```

```
$ root
```

```
root[] .demos
```

```
root[] .q (root の終了)
```

- ・次回のログイン時に自動的に設定が有効となるように、エディタで.bashrc を編集する。

```
$ emacs ~/.bashrc
```

- ・表示されたファイルの最後に、次の 1 行を追加する。({user}は自分のユーザ名)

```
cd /home/{user}/root/root-6.30.04-build ; . ./bin/thisroot.sh ; cd ~
```


・一度 WSL をログアウトしてから、再度 WSL にログインする。ログイン時にエラーが表示されないか、そしてログイン後に `root` が実行できるかを確認する。

```
$ exit
> wsl ~
(エラーが表示されないかを確認する)
$ root
root[] .demos
(実行できるかを確認する)
root[] .q
```

6) Windows のアプリをインストールする [オプション]

ここでインストールするアプリはオプションです。作業がしやすくなるので参考に紹介します。(Visual Studio Code はインストールしておくの良いと思います。その他は、使用環境によっては不要です。)

6-1) Visual Studio Code

プログラムコードの編集等を行うためのアプリです。emacs は嫌だという場合は、こちらを使うのも有りです。設定がされていれば、Ubuntu 上でも、次に様なコマンドで起動できます。

```
$ code {ファイル名}
```

下記の Web ページを参照してダウンロードとインストールを行ってください。

Visual Studio Code (<https://azure.microsoft.com/ja-jp/products/visual-studio-code/>)

参考 : WLS2 の Ubuntu のインストールについての補足

WSL でインストールをやり直したり、別のバージョンの Ubuntu を導入したいときのために、wsl に関するコマンドをいくつか紹介します。

- インストールした Ubuntu の登録抹消する。
 - > wsl --unregister Ubuntu
 - *完全な削除は、Windows の「設定」→「アプリ」で行う。
- インストール可能な OS を確認する
 - > wsl --list --online
- Ubuntu を再インストールする（別のバージョンを追加して導入する場合も可）
 - > wsl --install -d Ubuntu （現在は、こちらも Ubuntu-22.04 です）
 - > wsl --install -d Ubuntu-22.04
- 起動するデフォルト OS を設定（選択した OS の方が起動するようになります）
 - > wsl --set-default Ubuntu-22.04
- 起動するデフォルト OS の確認
 - > wsl --list

以上