

# Fine Kernel ToolKit システム (Windows C++ 版) セットアップマニュアル

## FineKernel Project (2024/4/18 版)

### 1 動作環境

Windows における Fine Kernel ToolKit システム C++ 版 (以下 FK) の動作環境は以下のとおりである。

- Windows10 64bit または Windows11 64bit
- OpenGL 4.1 以上の GPU サポート

そのため、macOS 上や、VMWare, Parallels Desktop などの仮想環境上では現時点では動作しない。Mac を利用する場合は Boot Camp で Windows10 をインストールし、Windows を起動して利用する必要がある。

### 2 Visual Studio のバージョンについて

現時点で FK は Visual Studio 2022 をサポートしている。本マニュアルでは、Visual Studio 2022 Community を前提に記述していくが、Professional, Enterprise でも手順は同様である。

### 3 前準備

インストールの前に、まず Windows Update をかけて OS を最新の状態にしておくこと。また、最低でも 12G 程度はドライブに空きを作っておく必要がある。

Visual Studio 2022 (以下「VS」) がインストールされている場合は、「Visual Studio Installer」を起動し、VS を最新版にアップデートしておくこと。未インストールの場合は、次節を参照しインストールを実施する。

### 4 VS2019 Community のインストール

VS Community のインストール方法は以下の通りである。

1. まず、下記 URL にアクセスする。

<https://visualstudio.microsoft.com/ja/downloads/>

2. ページ内にある Visual Studio 2022 欄内の「コミュニティ」(または「Community」)にある「無料ダウンロード」をクリックする。
3. ダウンロードされるインストーラを起動する。
4. 表示されるダイアログから「C++によるデスクトップ開発」のチェックを入れる。その他の項目については FK での開発には必要ないが、適宜自身が必要と思う項目を追加しておくことよい。
5. 本体のインストールが終わるまで待つ。数十分を要する。
6. 再起動が必要というダイアログが出た場合は、一度再起動する。
7. マイクロソフトアカウントを取得していない場合、事前に取得しておく。アカウントの設定を行わなくてもその時点では使用できるが、30日で使用できなくなるので必ず取得して Visual Studio 内で登録しておくこと。
8. VS を起動する。初回の起動は数分かかることがあるので、しばらく待つ。

9. 起動を確認したら終了する。

## 5 マイクロソフトアカウントの設定

VS は、利用の際にマイクロソフトアカウントでサインインをしておく必要がある。これは、大学で配付されている Office365 用アカウントで行うことを推奨するが、別のアカウントでも問題はない。

新規にマイクロソフトアカウントを作成する場合は、まず <https://account.microsoft.com> にアクセスし、左上にある「サインイン」を選び、そこで「作成」を選択する。

## 6 FK C++ 版のセットアップ

VS のインストールが完了したら、FK のセットアップを以下の手順で行う。

Windows 上の C++ で FK のプログラミングを行うには、FK VC++ 版をインストールする必要がある。まず、以下の URL の「インストーラーダウンロード」の欄にある、「Windows C++ 版」項目中のインストーラーをダウンロードする。

<https://gamescience.jp/FK>

ダウンロードの際、「ダウンロードしたユーザー数が少ないため、PC に問題を起こす可能性があります。」というメッセージが表示される場合があるが、問題ないのでそのまま継続していくこと。

このインストーラーを実行しようとする、「Windows によって PC が保護されました」というメッセージが表示される場合があるが、その場合は「詳細情報」をクリックして「実行」を選択する。あとは基本的に指示通りにインストールを進めればよい。

FK のシステムファイルがインストールされる箇所は、デフォルトでは「C:\FK\_VC22」となっている。この場所を変更する場合、以下の条件を満たす必要がある。

- ネットワークドライブ (GoogleDrive や OneDrive 等) は不可。
- 上位フォルダの名前に日本語が含まれている場合は不可。

インストールが終わった後、一度サインアウトか再起動を行っておくこと。

## 7 プロジェクトの作成方法

VS では、アプリケーションを作成する際には「プロジェクト」を作成する必要がある。開発には単なるプログラムソースだけでなく、様々な開発用の設定ファイルが必要となるが、プロジェクトはそれらを全て引くくめたものだと考えれば良い。

通常、プロジェクトには様々な設定を行う必要があるが、FK のセットアップが正常に完了した場合は、「プロジェクトウィザード」を利用することで簡易に設定を済ませることができる。以下にその手順を述べる。

1. VS を起動し、「新しいプロジェクトの作成」を選択する。
2. 列挙されているテンプレートの中から「FK\_C++.Template」を選択し「次へ」を押す。(大抵の場合、リストの最後に配置されている。)
3. 「プロジェクト名」に作成したいプロジェクトの名前を入力し、「場所」にプロジェクト全体を置きたいフォルダを指定し、「作成」を押す。数秒から数十秒程度でプロジェクトが作成される。「場所」はこのプログラムの置き場所になるので、個人の授業用フォルダなどを適切に選択しておく。ただし、以下の条件を満たす必要がある。
  - (a) ネットワークドライブ上 (GoogleDrive や OneDrive 等) のフォルダは不可。

(b) 上位フォルダの名前に日本語が用いられている場所は不可。

4. 「表示」メニューから「ソリューションエクスプローラー」を選び、「ソリューションエクスプローラー」ダイアログを表示しておく。この中の「ソースファイル」ある「main.cpp」をダブルクリックすると、サンプルコードを編集することができる。

この手順で、プロジェクトの作成と設定が完了する。実際にはプロジェクト自体に加えて、その上位に位置する「ソリューション」と呼ばれるファイルも同時に作成される。VS で作成したプロジェクトを開く場合は、ソリューションを通じて行う。

## 8 コンパイル (ビルド)

ソースファイルが作成できたらコンパイルを行い、実行ファイル (exe) を生成する。VS においては、コンパイルを初めとした一連の作業のことを「ビルド」と呼ぶ。ビルドを行うには、メニュー中の「ビルド」→「ソリューションのビルド」を選択する。

エラーが発生した場合はメッセージを確認し、ソースファイルにおけるエラーの場合はそれを修正する必要がある。ヘッダファイルのインクルードや、リンクの時点でエラーが発生している場合は、これまでの設定に何らかの不備がある可能性が高いので、もう一度これまでの手順を確認すること。

## 9 実行

ビルドが正常に終了した場合、プロジェクトファイル (.vcxproj) と同じフォルダ内にある bin フォルダ中に実行ファイルが生成される。実行するには、メニューの「デバッグ」→「デバッグなしで開始」を選択する。

### 9.1 ファイル読み込み処理を行うプログラムを実行する場合

プロジェクトフォルダの中に「Resources」というフォルダが存在するが、その中に入れたファイルやフォルダはビルド時に自動的に exe ファイルと同一の箇所にコピーされる。そのため、プログラム中で入力する様々なデータファイルは Resources フォルダ中に入れておくとよい。

### 9.2 デバッグビルドとリリースビルドについて

FK を使用したプログラムは、通常時はリリースビルドで実行することを推奨する。プロジェクト作成後の初期状態ではデバッグビルドになっているが、この状態ではプログラムの処理速度が大幅に低下することがある。開発を行っていくうち、複雑なバグなどが発生した場合にはデバッグビルドで問題を検証するのが有効であるので、状況に応じて切り替えるとよい。

## 10 プロジェクトの保存とロード

プロジェクト全体の状態を保存するには、メニューの「ファイル」→「すべてを保存」を実行する。これにより、全てのソースファイルとプロジェクトの情報が保存される。

保存したプロジェクトをロードするには、“sln”を拡張子を持つファイルを開くか、VS を起動後メニュー中の「ファイル」→「ソリューションを開く」を選択し、“sln”を拡張子を持つファイルを選択すればよい。