

DirectX11授業用配布プロジェクト 開発マニュアル

作成日 : 2022.2.8
更新日 : 2024.3.12
作成者 : 中野敦史

● はじめに	2
● DirectX SDK 2010 June のインストール	2
○ DirectX SDK とは？	2
○ インストール方法	2
● DirectX SDKをプロジェクトに設定	3
○ インストールしただけでは使えない	3
○ 設定方法	3
■ インクルード ディレクトリの設定	3
■ ライブラリ ディレクトリの設定	3
● プリコンパイル済みヘッダーの設定	4
○ プリコンパイル済みヘッダーとは？	4
○ 設定方法	4
■ 「stdafx.h」の使用設定	4
■ 全てのcppファイルで「stdafx.h」の読み込み設定	5
■ 「stdafx.cpp」の使用設定	5
● Visual Studio 2019 以降でのWindowsアプリ開発の設定	6
○ デフォルト設定ではビルド通らない	6
○ Windows API 用のエントリーポイントを設定	6
● 終わりに	6

● はじめに

DirectX11基礎の授業で使用するプロジェクト「3D_Base」の設定について書かれたマニュアルである。開発するにあたって最初に必要な設定をまとめてある。うまくできない場合は連絡ください。

● DirectX SDK 2010 June のインストール

○ DirectX SDK とは？

授業で使用するDirectXのバージョンは、「11(イレブン)」になる。

インストールしてもらうのは「9(ナイン)」になる。

授業では「Xファイル (.x) 」の3Dデータを使用した開発を行っている。

3Dデータの読み込み時にDirectX9の一部機能を使用しているのでインストールが必要になる。

○ インストール方法

下記のGoogleの共有ドライブにマニュアルがあるので手順にしたがってインストールする。

- ・ゲーム専攻 配布用

- ↳ソフトウェア一覧

- ↳DirectXSDK

- ↳DirectX SDK インストールマニュアル.pdf

● DirectX SDKをプロジェクトに設定

○ インストールだけでは使えない

Visual Studioを開き、プロジェクトの設定が必要になる。

この作業は、新しくプロジェクトを立ち上げるたびに必要になる。

※最初の一度しか設定しないので、設定をすること自体を忘れがち。覚えておくこと。

○ 設定方法

■ インクルード ディレクトリの設定

- ①「ソリューションエクスプローラー」の「3D_Base」を右クリックする。
- ②「プロパティ」をクリックで開く。
- ③「構成」を「すべての構成」に変更する。
- ④「プラットフォーム」を「x64」に変更する。
- ⑤少し下の「構成のプロパティ」の「VC++ ディレクトリ」をクリックする。
- ⑥右側の項目の「全般」にある「インクルードディレクトリ」をクリックする。
- ⑦右端に「v」が出るのでクリックし、「編集」をクリックする。
- ⑧空欄に「\$(DXSDK_DIR)Include」を入力して「OK」をクリックする。
- ⑨右下の「適用」をクリックする。

■ ライブラリ ディレクトリの設定

上記「インクルードファイルの設定」の①～⑤まで進める。

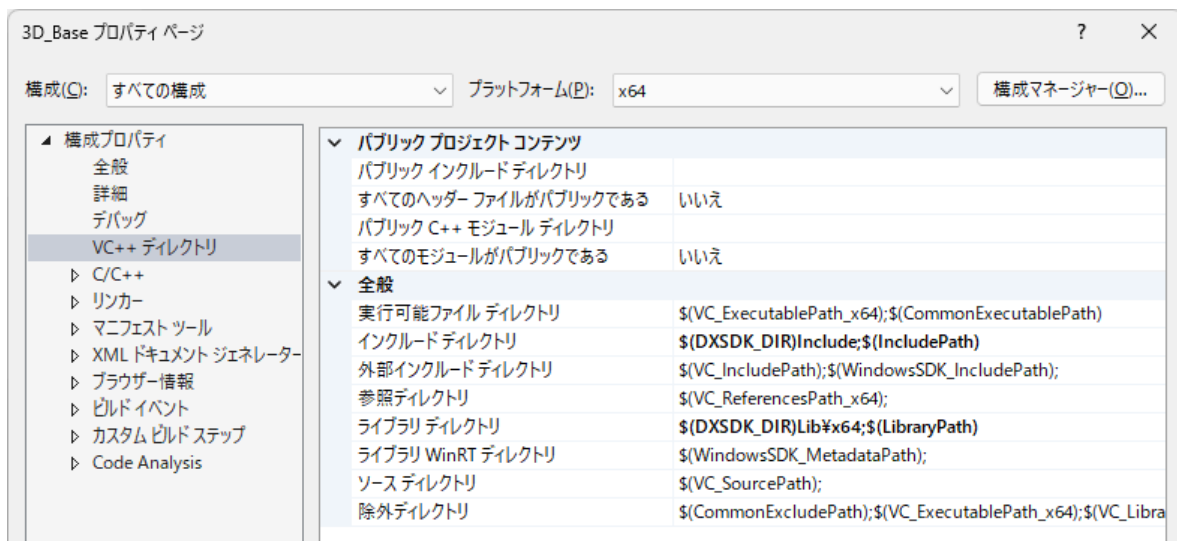
- ⑥右側の項目の「全般」にある「ライブラリ ディレクトリ」をクリックする。

- ⑦下向きの「<」が出るのでクリックし、「編集」をクリックする。

- ⑧空欄に「\$(DXSDK_DIR)Lib¥x64」を入力して「OK」をクリックする。

※「¥」は半角で入力すること。

- ⑨右下の「適用」をクリックし、「OK」をクリックする。



● プリコンパイル済みヘッダーの設定

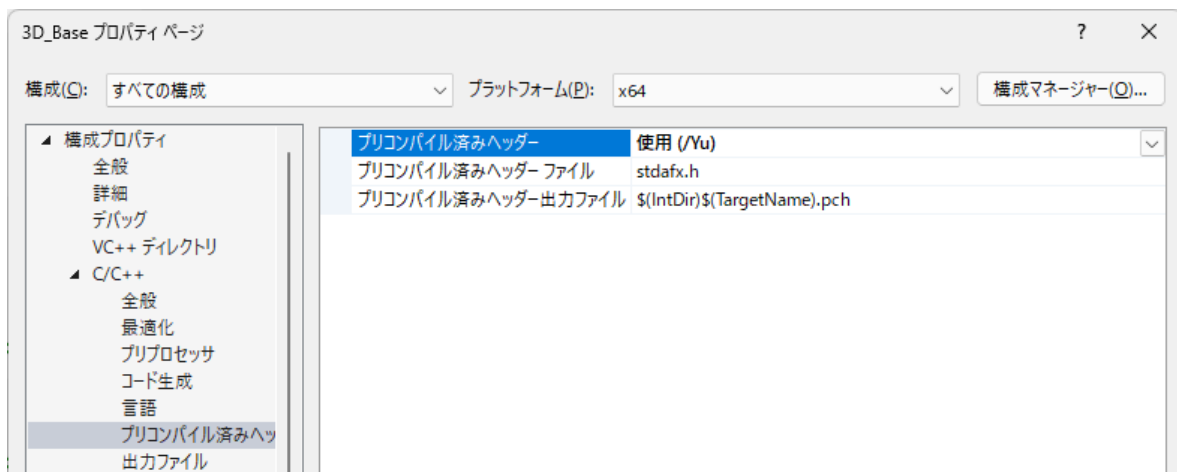
○ プリコンパイル済みヘッダーとは？

通常いろいろな場所で同じヘッダーファイルを読み込むと、そのたびに解析されてビルドが遅くなる。滅多に変更しないが、使用頻度の高いヘッダーファイルのインクルード時間を節約する機能がある。また、設定にもよるが、`#include` “プリコンパイル済みヘッダー” 以前には、コメント以外のコードは書かないようにすること。

○ 設定方法

■ 「stdafx.h」の使用設定

- ①「ソリューションエクスプローラー」の「3D_Base」を右クリックする。
- ②「プロパティ」をクリックで開く。
- ③「構成」を「すべての構成」に変更する。
- ④「プラットフォーム」を「x64」に変更する。
- ⑤少し下の「構成のプロパティ」の「C/C++」の▷をクリックする。
- ⑥「プリコンパイル済みヘッダー」をクリックする。
- ⑦「プリコンパイル済みヘッダー」を「使用(/Yu)」に変更する。
- ⑧右下の「適用」をクリックし、「OK」をクリックする。



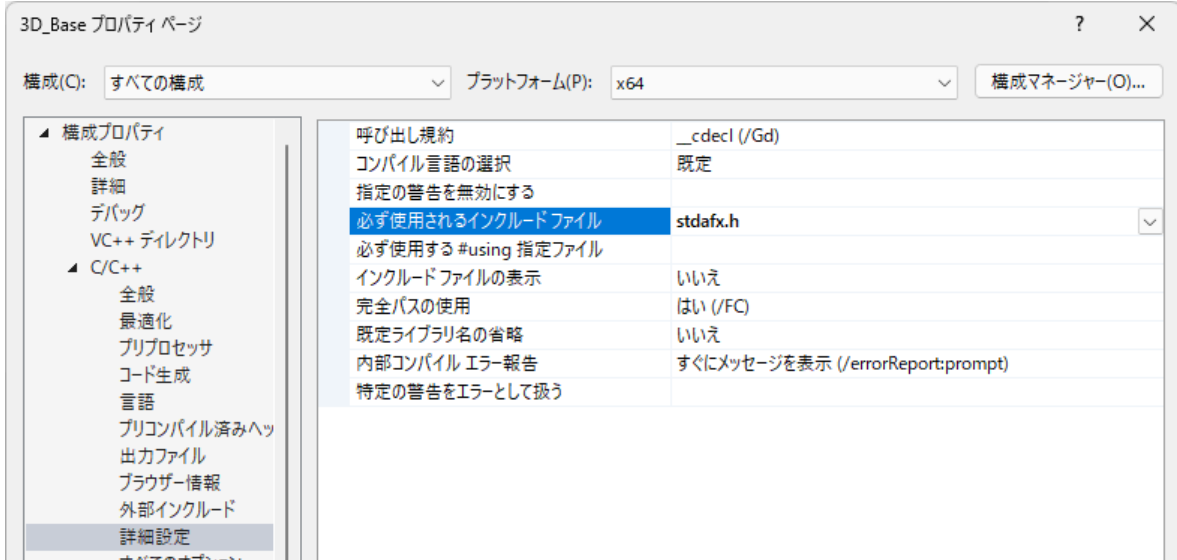
- 全てのcppファイルで「stdafx.h」の読み込み設定

上記【「stdafx.h」の仕様設定】の①～⑤まで進める。

⑥「詳細設定」をクリックする。

⑦「必ず使用されるインクルード ファイル」の空欄に「stdafx.h」と入力する。

⑧右下の「適用」をクリックし、「OK」をクリックする。



■ 「stdafx.cpp」の使用設定

①「ソリューションエクスプローラー」の「stdafx.cpp」を右クリックする。

②「プロパティ」をクリック。

③「構成」を「すべての構成」に変更する。

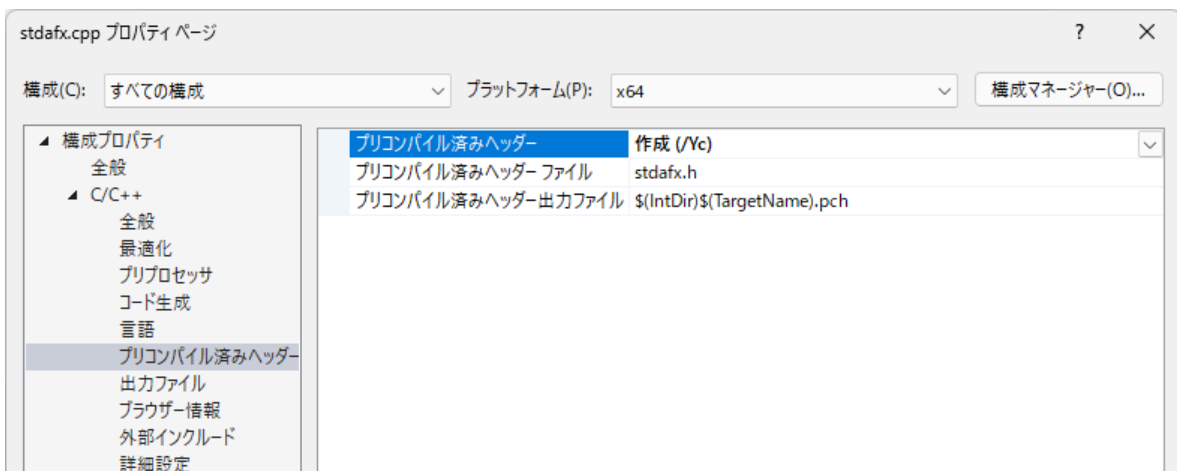
④「プラットフォーム」を「x64」に変更する。

⑤少し下の「構成のプロパティ」の「C/C++」の▷をクリックする。

⑥「プリコンパイル済みヘッダー」をクリックする。

⑦「プリコンパイル済みヘッダー」を「作成(Yc)」に変更する。

⑧右下の「適用」をクリックし、「OK」をクリックする。



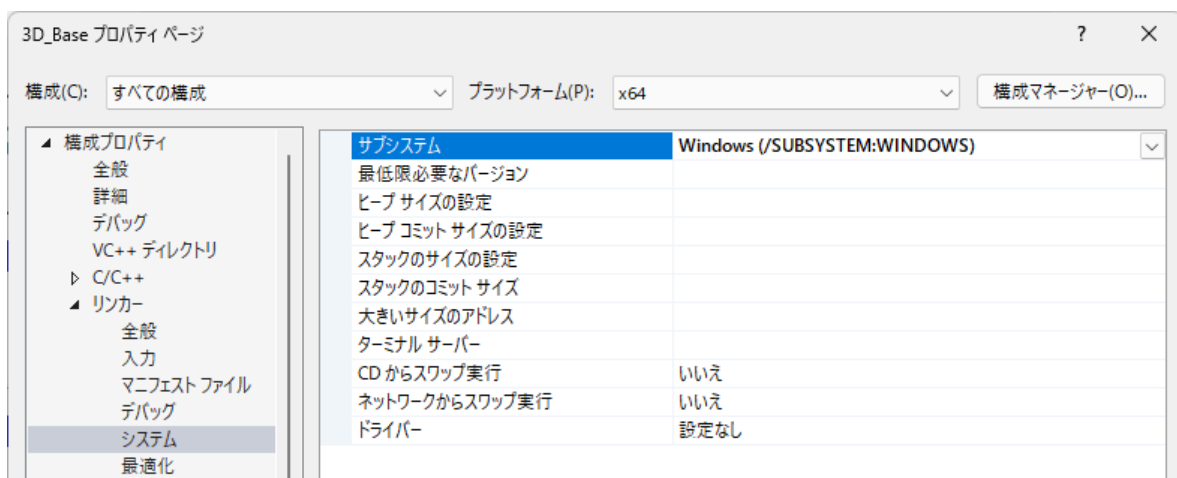
● Visual Studio 2019 以降でのWindowsアプリ開発の設定

○ デフォルト設定ではビルド通らない

Visual Studio 2019以降で「空のプロジェクト」から作成したプロジェクトのエントリーポイントが「main関数（コンソール）」になってしまっている。Windows API を使用するので、設定を変更する必要がある。

○ Windows API 用のエントリーポイントを設定

- ①「ソリューションエクスプローラー」の「stdafx.cpp」を右クリックする。
- ②「プロパティ」をクリック。
- ③「構成」を「すべての構成」に変更する。
- ④「プラットフォーム」を「x64」に変更する。
- ⑤少し下の「構成のプロパティ」の「リンカー」の▷をクリックする。
- ⑥「システム」をクリックする。
- ⑦「サブシステム」を「Windows(/SUBSYSTEM:WINDOWS)」に変更する。
※「設定なし」でも可。
- ⑧右下の「適用」をクリックし、「OK」をクリックする。



● 終わりに

ここまでの設定が全て完了し、ビルドが通ってようやく開発を始めることができます。
結構大変ですが、開発環境の構築とは、敷居が高いものです。
また、何かトラブルがあれば早急に連絡してください。