

2021 計算機プログラミング演習

Microsoft
Visual Studio 2019 Community
インストール手順

Visual Studio とは

- Windows OS 向けのソフトウェア統合開発環境
- 以下の豊富なプログラミング言語を提供
 - C / C++ 言語
 - C#
 - Visual BASIC
- 本授業／演習では C 言語を使用する

Visual Studio 2019 Community とは

- 学生や個人開発者は無償で利用可
 - ✓ サインインが必要
 - ✓ ネットワークからインストール
- PC(ハードウェアの要件)*
 - ✓ 1.8GHz以上のプロセッサ クアッドコア以上を推奨
 - ✓ 2GBのRAM(メモリ)が必要 8GBのRAM推奨
 - ✓ ハードディスクの空き領域は20から50GB必要

ハードディスクの空き領域など自分のPCの状況を調べて
問題なければ, Windows UpdateでWindows10を最新の状態
にした後にインストール作業にとりかかること！

* Visual Studio2019製品ファミリのシステム要件

<https://docs.microsoft.com/ja-jp/visualstudio/releases/2019/system-requirements>

インストールする前

- OSはWindows10にすること(Windows7は既にサポート終了しているので使わない)
- Windows Updateで最新の状態にすること
- ノート PC のAC アダプタを必ず使用する
- PC が再起動する場合があるので、作業中のアプリケーションは作業内容を保存して終了しておく

参考

生協で購入したPCでインストールを実施した例

インストールを試したノートPC: Fujitsu LAPTOP

- CPUクロック周波数 2.5GHz
- メモリ4GB
- 空きハードディスク容量 70GB（これは重要！！）

上記のスペックでインストールにかかった時間はおよそ18分程度だが、ネットワーク環境やPCのスペックにより大きく異なるので注意

Visual Studio Communityのダウンロード



ここをクリック

下記のMicrosoftのサイトからダウンロードできます
(Googleで"Visual Studio Community"と検索すると出てきます)

<https://visualstudio.microsoft.com/ja/vs/community/>

Visual Studio Communityのインストール



ダウンロードすると自分のPCのダウンロードフォルダの中に” vs_community__ (ここには数字が入る).exe” という実行ファイルがある

- ① 実行ファイルをクリック
- ② 「このアプリがデバイスに変更を加えることを許可しますか？」 “はい”を選択
- ③ “続行”を選択

PC環境によっては下記の④, ⑤, ⑥, ⑦の手順がある

- ④ 更新プログラムが利用可能(バージョン16.9.3) “更新”を選択
- ⑤ 再起動が必要 “再起動”を選択
- ⑥ デバイスのセットアップを完了しましょう “続行”を選択
- ⑦ 推奨ブラウザの設定を使用 “今はスキップ”を選択

Visual Studio Communityのインストール

⑧ Visual Studio installerが自動で立ち上がる 立ち上がらない場合はWindowsメニューにあるVisual Studio installerをクリック

インストールしています - Visual Studio Community 2019 - 16.9.3

ワークロード 個別のコンポーネント 言語パック インストールの場所

インストールする項目の選択でヘルプが必要ですか? [詳細情報](#)

 **Python 開発**
Python の編集、デバッグ、対話型開発、ソース管理。

☐

 **Node.js 開発**
Node.js (非同期イベントドリブ JavaScript ランタイム) を使用してスケーラブルなネットワーク アプリケーションをビルドします。

☐

デスクトップとモバイル (5)

 **.NET デスクトップ開発**
.NET Core と .NET Framework と共に C#, Visual Basic, F# を使用して、WPF、Windows フォーム、コンソール アプリケーションを...

☐

 **C++ によるデスクトップ開発**
MSVC、Clang、CMake、MSBuild など、選択したツールを使用して、Windows 用の最新の C++ アプリをビルドします。

☐

 **ユニバーサル Windows プラットフォーム開発**
C#, VB, または C++ (オプション) を使ってユニバーサル Windows プラットフォームのアプリケーションを作成します。

☐

 **.NET によるモバイル開発**
Xamarin を使用して iOS、Android、または Windows 向けのクロスプラットフォーム アプリケーションをビルドします。

☐

場所
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community [変更...](#)

続行すると、選択した Visual Studio のエディションの [ライセンス](#) に同意することになります。また、Visual Studio を使用して他のソフトウェアをダウンロードする機能も提供されます。このソフトウェアは、[サードパーティに関する通知](#) または付属するライセンスに記載のとおり、個別にライセンスされています。続行することで、これらのライセンスにも同意することになります。

必要な領域の合計 861 MB

ダウンロードしながらインストールする

Visual Studio Communityのインストール

⑨ 「C++によるデスクトップ開発」を選択しインストール

インストールしています - Visual Studio Community 2019 - 16.9.3

ワークロード 個別のコンポーネント 言語パック インストールの場所

インストールする項目の選択でヘルプが必要ですか? 詳細情報

Python 開発
Python の編集、デバッグ、対話型開発、ソース管理。

Node.js 開発
Node.js (非同期イベント ドリブ JavaScript ランタイム) を使用してスケーラブルなネットワーク アプリケーションをビルドします。

デスクトップとモバイル (5)

.NET デスクトップ開発
.NET Core と .NET Framework と共に C#, Visual Basic, F# を使用して、WPF、Windows フォーム、コンソール アプリケーションを...

C++ によるデスクトップ開発
MSVC、Clang、CMake、MSBuild など、選択したツールを使用して、Windows 用の最新の C++ アプリをビルドします。

ユニバーサル Windows プラットフォーム開発
C#、VB、または C++ (オプション) を使ってユニバーサル Windows プラットフォームのアプリケーションを作成します。

.NET によるモバイル開発
Xamarin を使用して iOS、Android、または Windows 向けのクロスプラットフォーム アプリケーションをビルドします。

インストールの詳細

C++ によるデスクトップ開発

- 含む
 - ✓ C++ コア デスクトップ機能
- オプション
 - ✓ MSVC v142 - VS 2019 C++ x64/x86 ビルド...
 - ✓ Windows 10 SDK (10.0.19041.0)
 - ✓ Just-In-Time デバッガー
 - ✓ C++ のプロファイル ツール
 - ✓ Windows 用 C++ CMake ツール
 - ✓ 最新の v142 ビルド ツール用 C++ ATL (x86 お...
 - ✓ Boost.Test のテスト アダプター
 - ✓ Test Adapter for Google Test
 - ✓ Live Share
 - ✓ IntelliCode
 - ✓ C++ AddressSanitizer
 - ☐ 最新の v142 ビルド ツール用 C++ MFC (x86...
 - ☐ v142 ビルド ツール用 C++/CLI サポート (最新)
 - ☐ v142 ビルド ツール用 C++ モジュール (x64/x86...
 - ☐ Windows 用 C++ Clang ツール (11.0.0 - x64...
 - ☐ JavaScript 診断

場所
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community 変更...

続行すると、選択した Visual Studio のエディションの [ライセンス](#) に同意することになります。また、Visual Studio を使用して他のソフトウェアをダウンロードする機能も提供されます。このソフトウェアは、[サードパーティに関する通知](#) または付属するライセンスに記載のとおり、個別にライセンスされています。続行することで、これらのライセンスにも同意することになります。

必要な領域の合計 7.46 GB

ダウンロードしながらインストールする ▾ **インストール(I)**

チェックを入れる



インストールをクリック

Visual Studio Communityのインストール

インストール中の画面

Visual Studio インストーラー

インストール済み 使用可能



Visual Studio Community 2019

一時停止(P)

ダウンロードおよび検証中: 1.96 GB の 502 MB

(28 MB/秒)

24%

インストール中: 395 個中 31 個のパッケージ

3%

Microsoft.VisualStudio.Navigation.RichCodeNav

☒ インストール後に起動する

[リリースノート](#)

Visual Studio Communityの使用

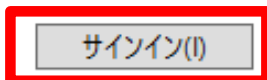
Visual Studio

Visual Studio にサインイン

- デバイス間での設定の同期
- Live Share を使用してリアルタイムで共同作業を行う
- Azure サービスとシームレスに統合する

サインインしない場合、試用期間が過ぎると使えなくなるので注意！！

ここをクリック



アカウントがありませんか？ 作成してください。

後で行う。

⑩ スタートメニューにVisual Studio 2019があることを確認してクリック

⑪ Visual Studioにサインインが必要
“サインイン”を選択

⑫ Microsoftのアカウントがない場合は作成



サインイン

メール、電話、Skype

アカウントをお持ちではない場合、作成できます。

アカウントにアクセスできない場合

アカウントがない場合は作成



戻る

次へ

Visual Studio Communityの使用

Visual Studio 2019

最近開いた項目(R)

Visual Studio を使用するとき、ユーザーが開くプロジェクト、フォルダー、ファイルはここに表示されるので、すばやくアクセスできます。

頻繁に開く項目は、ピン留めして常に一覧の先頭に表示することができます。

開始する



コードスペースに接続(E)

クラウドを利用した開発環境を作成および管理します



リポジトリのクローン(C)

GitHub や Azure DevOps などのオンライン リポジトリからコードを取得します



プロジェクトやソリューションを開く(P)

ローカルの Visual Studio プロジェクトまたは .sln ファイルを開きます



ローカル フォルダーを開く(F)

任意のフォルダー内のコードに移動して編集します



新しいプロジェクトの作成(N)

開始するには、コード スキャフォールディング付きのプロジェクト テンプレートを選択します

[コードなしで続行\(W\) →](#)

新しいプロジェクト
の作成(N)を選択

Visual Studio Communityの使用

新しいプロジェクトの作成

最近使用したプロジェクト テンプレート(R)

最近アクセスしたテンプレートの一覧は、ここに表示されます。

空の新しいプロジェクト
を選択

テンプレートの検索(S) (Alt+S)

すべての言語(L)

すべてのプラットフォーム(P)

すべてのプロジェクトの種類...



空のプロジェクト

Windows 用に C++ で最初から始めます。開始ファイルを提供しません。

C++

Windows

コンソール



コンソール アプリ

Windows ターミナルでコードを実行します。既定では "Hello World" を出力します。

C++

Windows

コンソール



Windows デスクトップ ウィザード

ウィザードを使用して、独自の Windows アプリを作成します。

C++

Windows

デスクトップ

コンソール

ライブラリ



Windows デスクトップ アプリケーション

Windows 上で実行されるグラフィカル ユーザー インターフェイス付きアプリケーションのプロジェクトです。

C++

Windows

デスクトップ



空のソリューション

プロジェクトを含まない空のソリューションを作成します。

その他



ダイナミック リンク ライブラリ (DLL)

戻る(B)

次へ(N)

Visual Studio Communityの使用

新しいプロジェクトを構成します

空のプロジェクト C++ Windows コンソール

プロジェクト名(N)

Myname1

ここは Myname1とする

場所(L)

C:\Users\Nakano\source\ProjectsNakano

「Projects名前」の名前の部分にはあなたの名前を半角英
字で指定してください。(ここでは ProjectsNakano)

ソリューション名(M) ⓘ

Myname1

ここは Mynameとする

☐ ソリューションとプロジェクトを同じディレクトリに配置する(D)

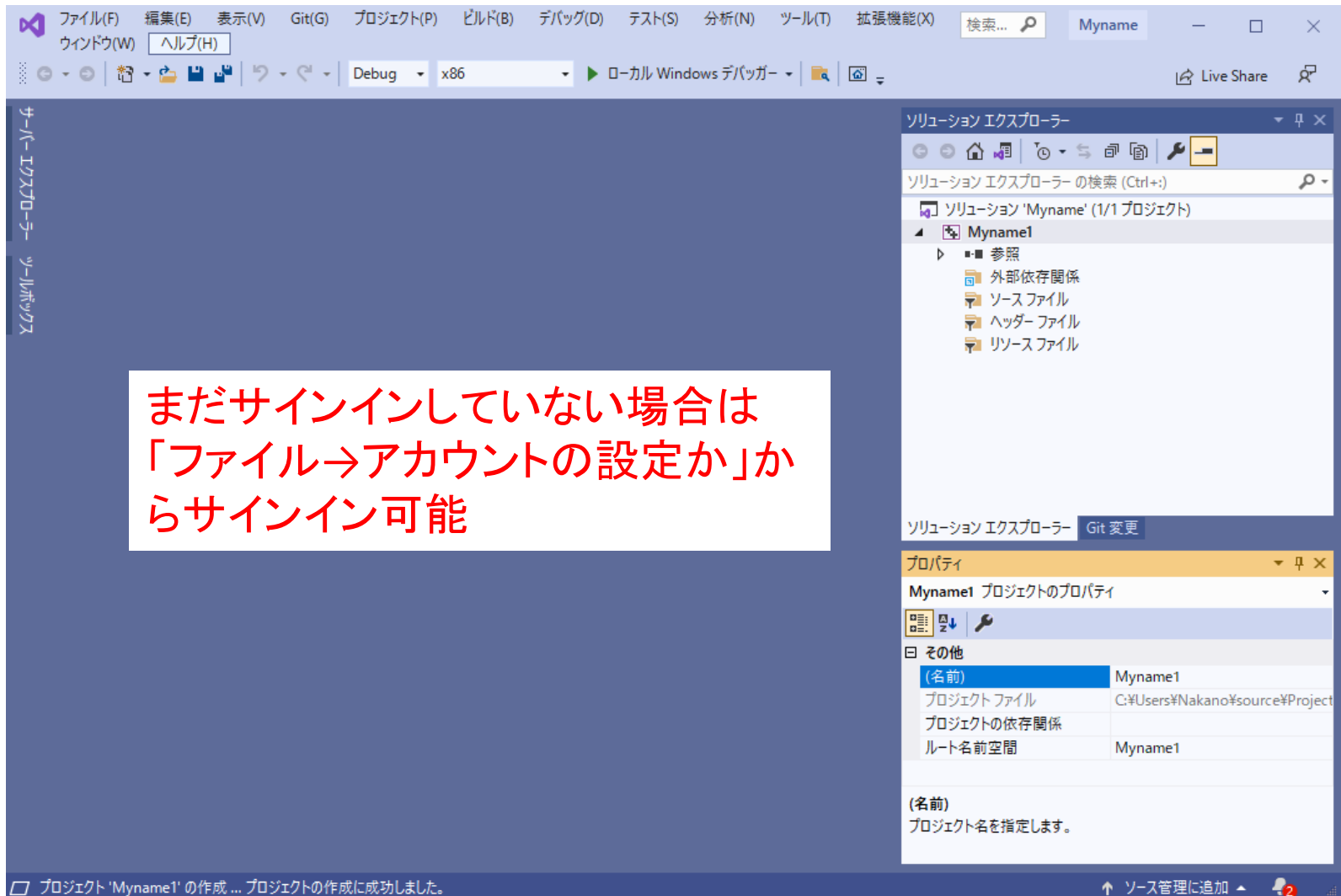
作成をクリック

戻る(B)

作成(C)

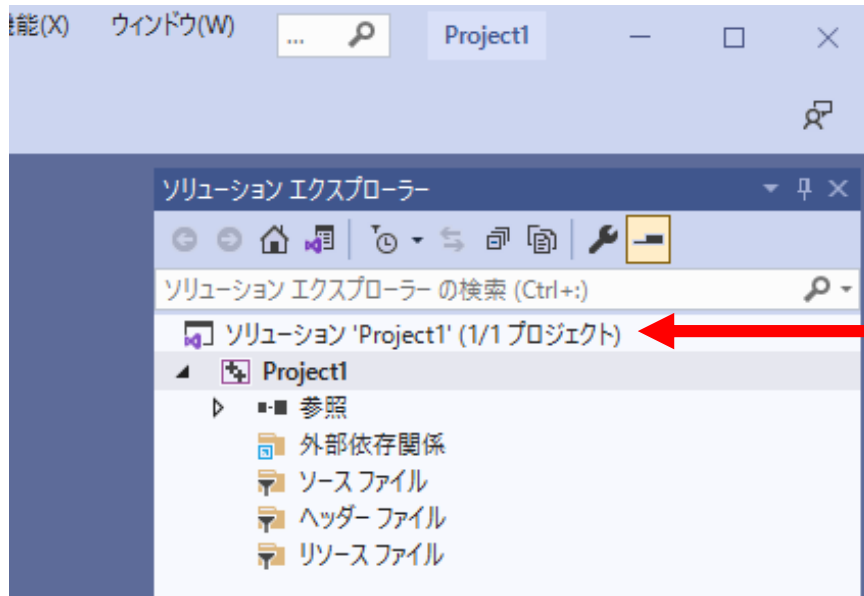
Visual Studio Communityの使用

空のプロジェクトなのでまだ何もない状態



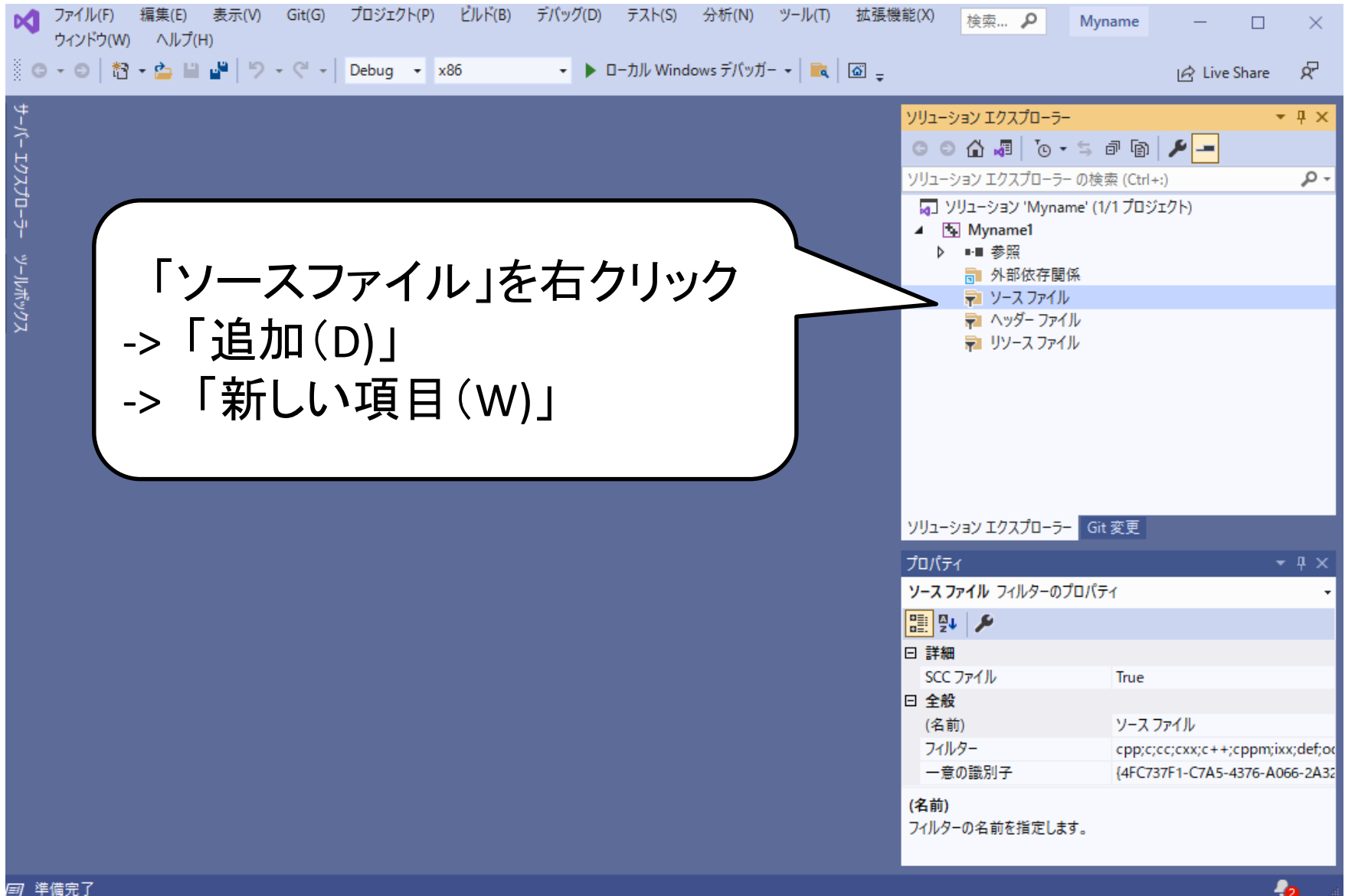
Visual Studio Communityの使用

(間違えた場合)ソリューション名を訂正したい



ここにカーソルを合わせて右クリック
「名前の変更(M)」を選択して訂正

Visual Studio Communityの使用



C ソースファイルの作成

新しい項目の追加 - Myname1

インストール済み

並べ替え: 既定

検索 (Ctrl+E)

Visual C++

- コード
- 書式設定
- ATL
- データ
- リソース
- Web
- ユーティリティ
- プロパティシート
- HLSL
- Test
- グラフィックス

オンライン

種類: Visual C++
C++ ソースコードを含むファイルを作成します

① 「C++ ファイル (.cpp)」を選択

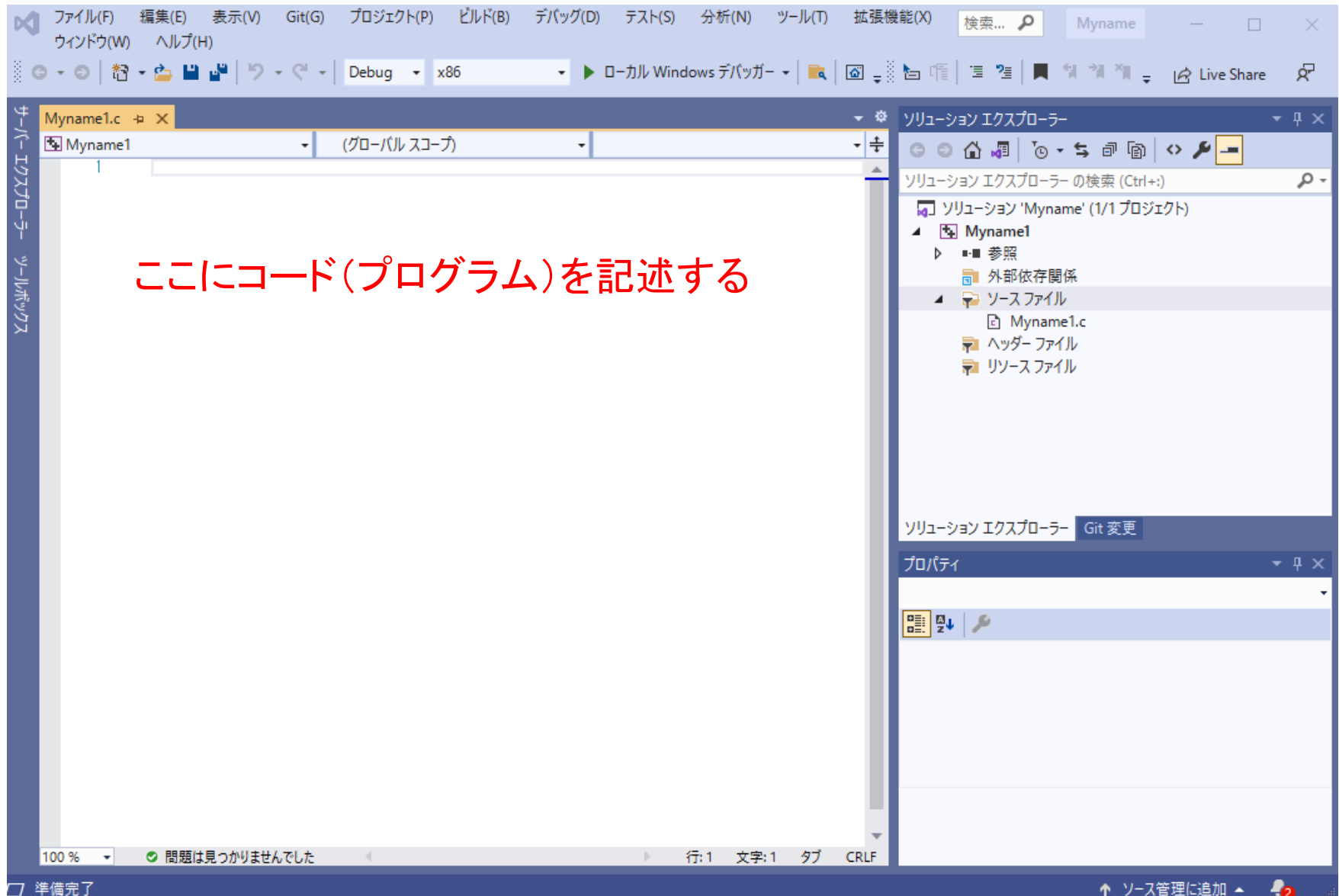
② 「Myname1.c」と入力
拡張子 .c を忘れないように！

名前(N): Myname1.c
場所(L): C:\Users\Nakano\source\Projects\Nakano\My

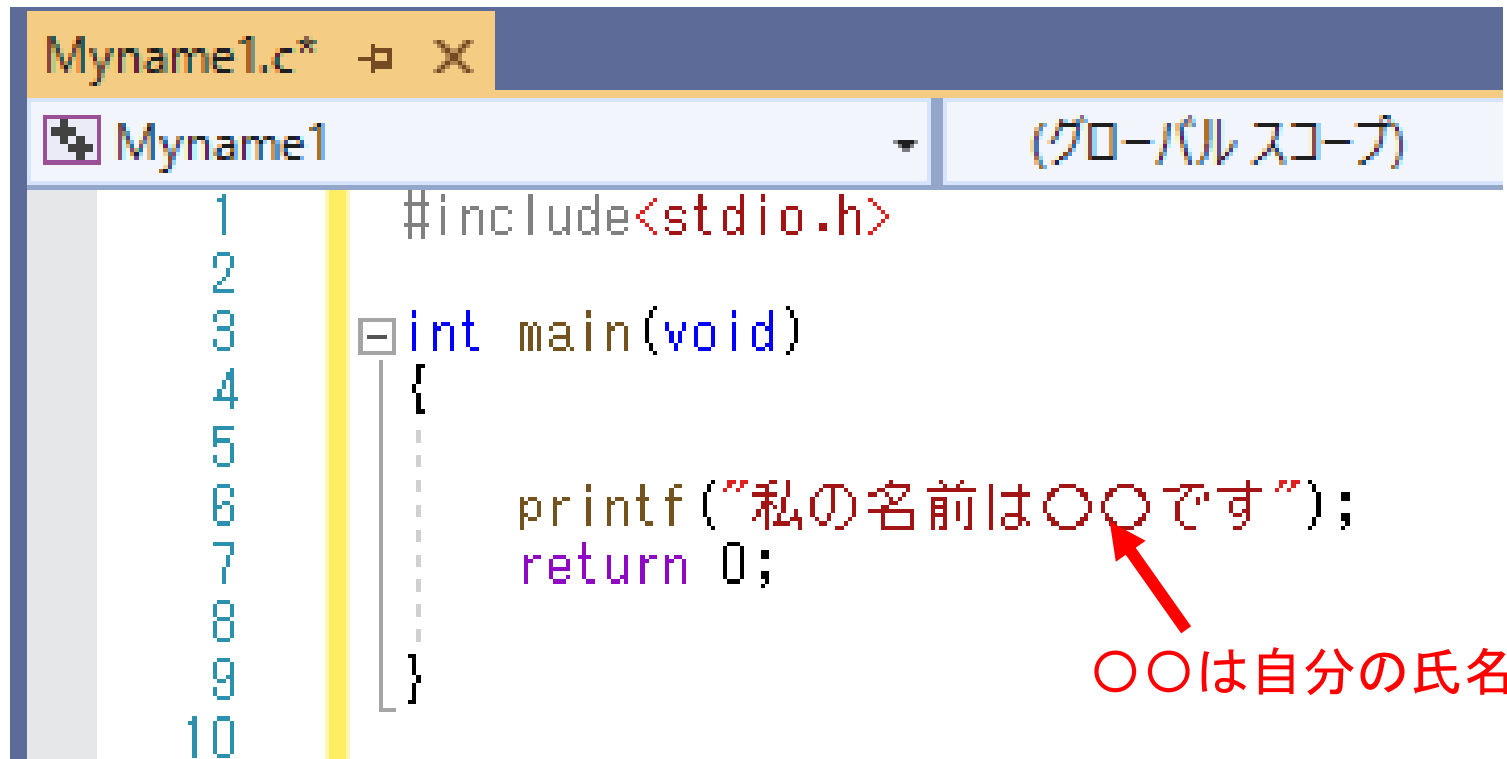
③ 「追加」をクリック

追加(A) キャンセル

「MyName1.c」の編集画面



「MyName1.c」の編集画面

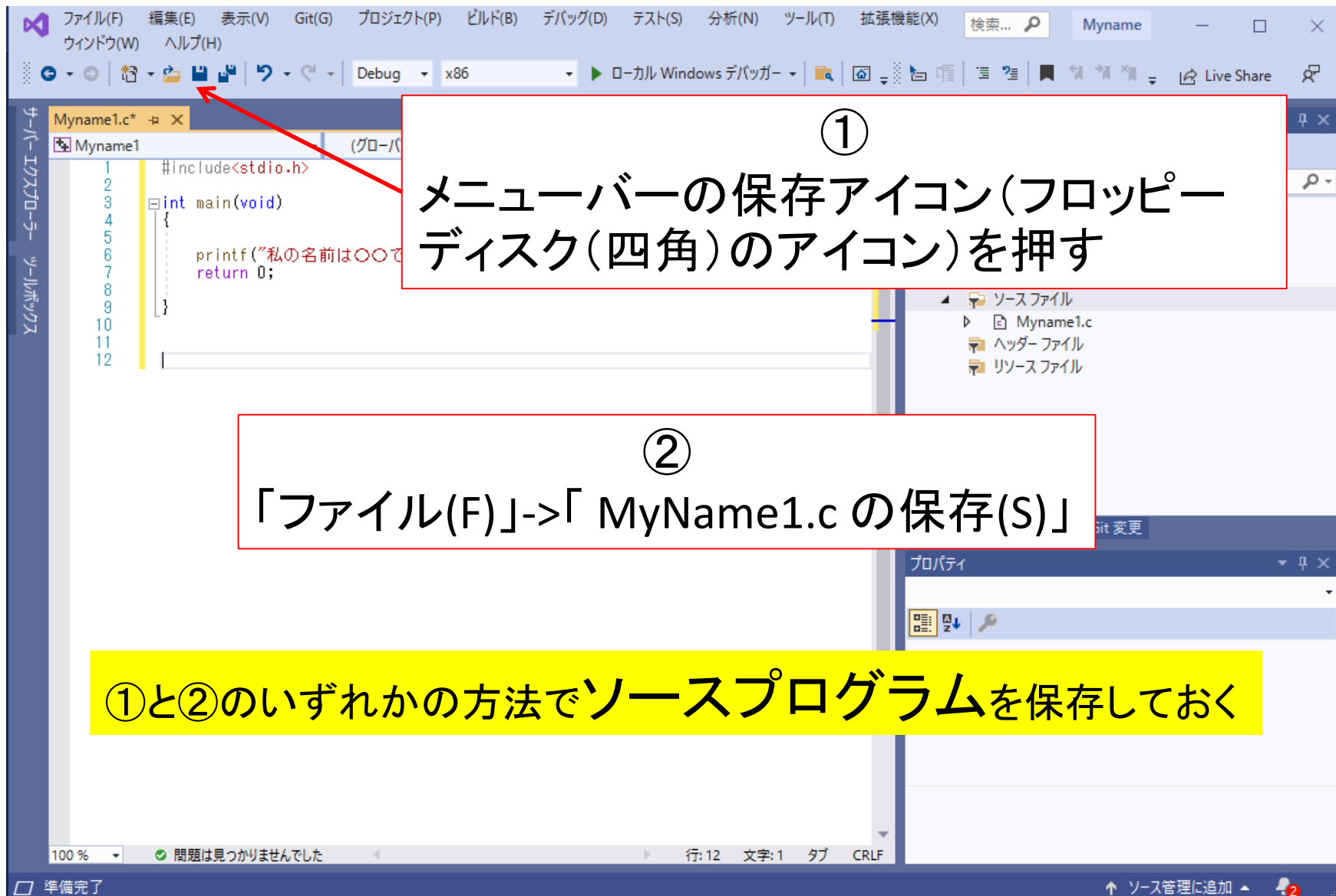


```
Myname1.c*  X
Myname1 (グローバルスコープ)
1  #include<stdio.h>
2
3  int main(void)
4  {
5
6      printf("私の名前は〇〇です");
7      return 0;
8  }
9
10
```

〇〇は自分の氏名

- キーボード入力モードが「半角英数」になっていること！
- 英字の小文字 (abc) で書く！ 大文字(ABC)はエラーになる
- Caps Lockに注意!!
- 6, 7行目にはTabを入れる

ソースプログラムの保存



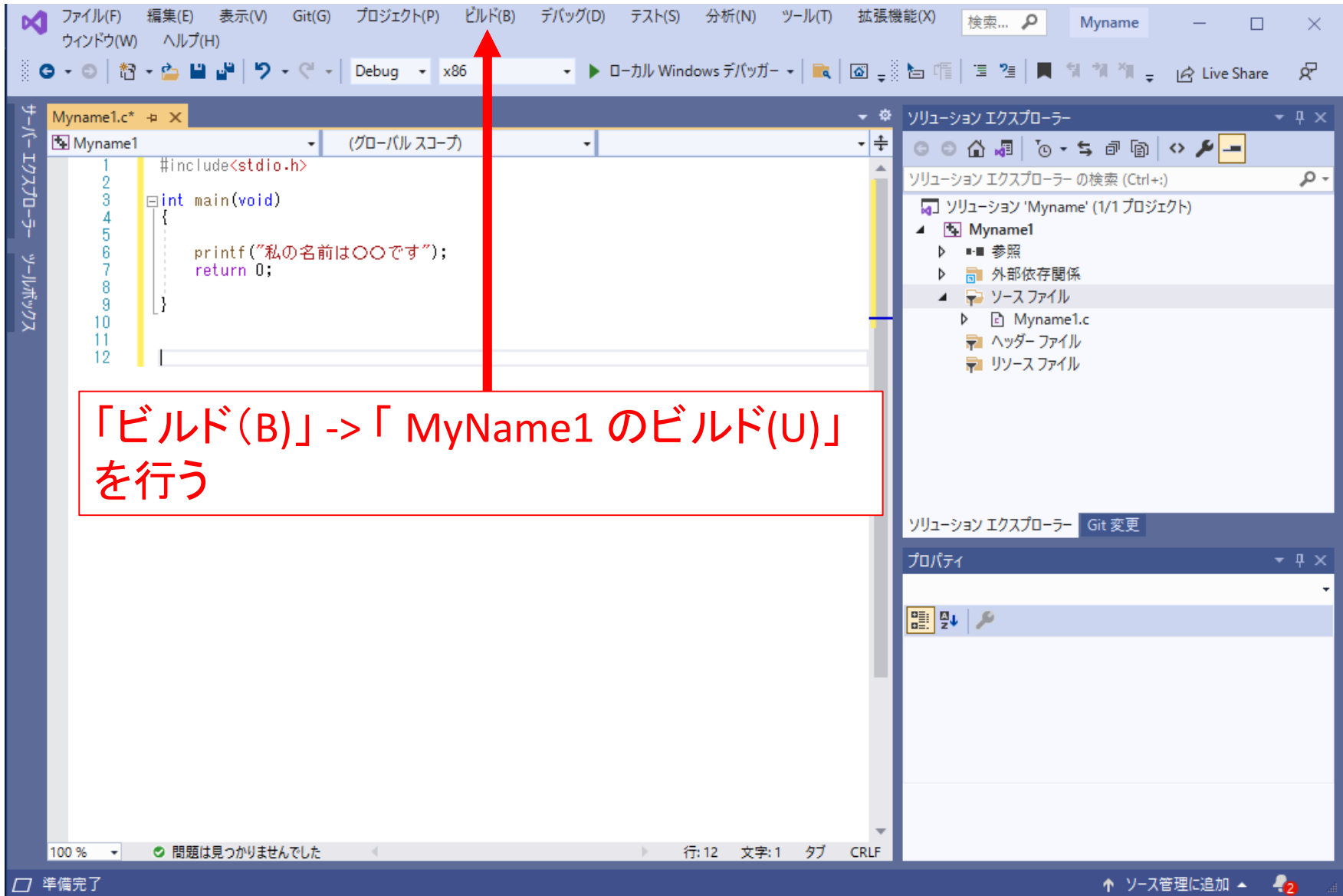
The screenshot shows the Visual Studio IDE with a C program named 'Myname1.c' open. A red arrow points to the save icon (floppy disk) in the toolbar, which is circled with a red line and labeled with a circled '1'. Below this, a text box explains that this is the save icon. Another text box, labeled with a circled '2', points to the 'File' menu and the 'Save' option. A yellow box at the bottom summarizes the two methods.

①
メニューバーの保存アイコン(フロッピーディスク(四角)のアイコン)を押す

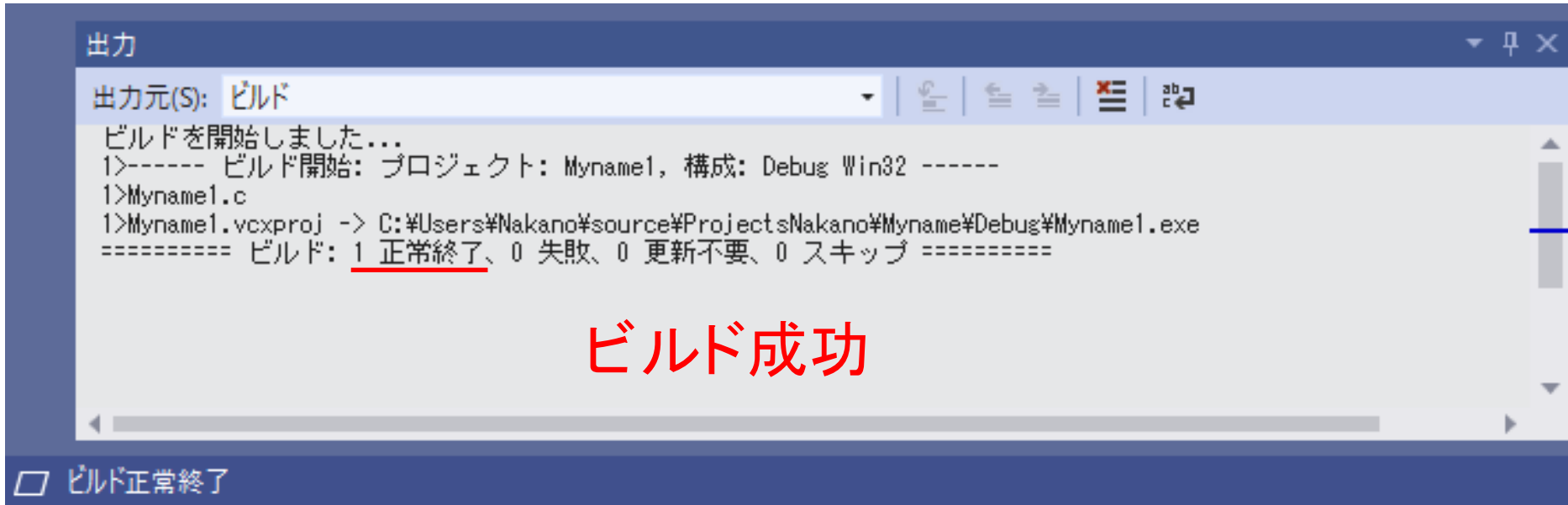
②
「ファイル(F)」->「 MyName1.c の保存(S)」

①と②のいずれかの方法でソースプログラムを保存しておく

「MyName1.c」のビルド



「MyName1.c」のビルド成功



The screenshot shows the 'Output' window in Visual Studio. The 'Output Source' is set to 'ビルド' (Build). The output text is as follows:

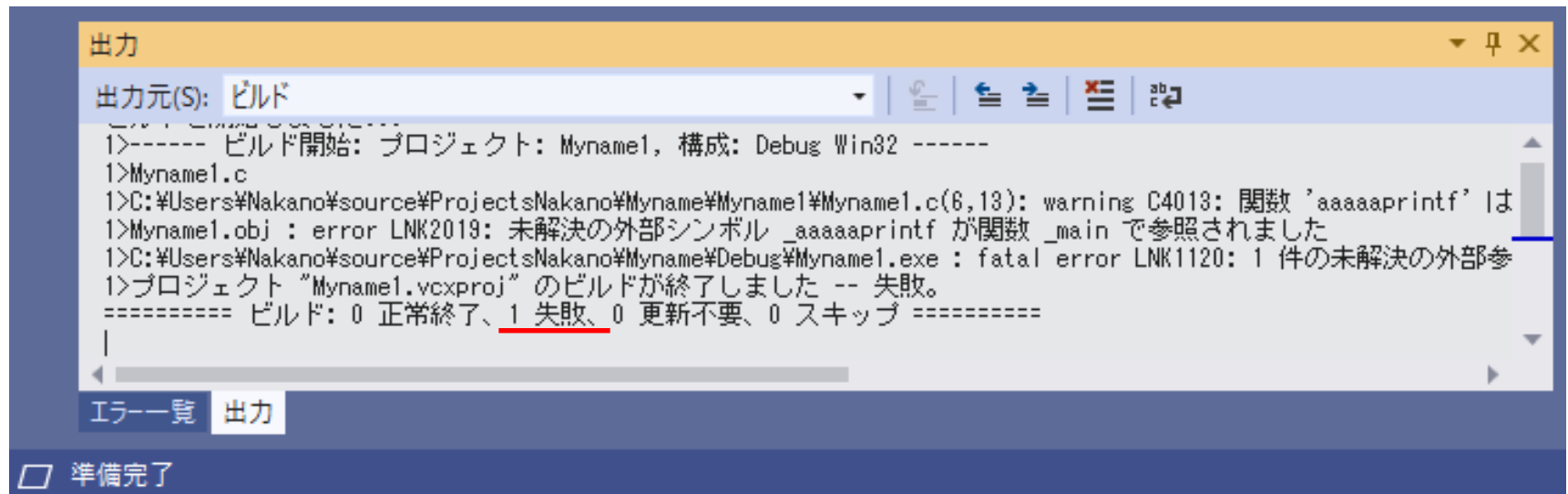
```
ビルドを開始しました...  
1>----- ビルド開始: プロジェクト: Myname1, 構成: Debug Win32 -----  
1>Myname1.c  
1>Myname1.vcxproj -> C:\Users\Nakano\source\Projects\Nakano\Myname\Debug\Myname1.exe  
===== ビルド: 1 正常終了, 0 失敗, 0 更新不要, 0 スキップ =====
```

Below the output text, the words 'ビルド成功' (Build Success) are written in large red characters.

At the bottom of the window, a status bar shows a green square icon followed by the text 'ビルド正常終了' (Build completed successfully).

プログラムの実行については次回行う

「MyName1.c」のビルド失敗



The screenshot shows the 'Output' window in Visual Studio. The 'Output Source' is set to 'Build'. The output text is as follows:

```
1>----- ビルド開始: プロジェクト: Myname1, 構成: Debug Win32 -----  
1>Myname1.c  
1>C:\Users\Nakano\source\Projects\Nakano\Myname\Myname1\Myname1.c(6,13): warning C4013: 関数 'aaaaaprintf' は  
1>Myname1.obj : error LNK2019: 未解決の外部シンボル _aaaaaprintf が関数 _main で参照されました  
1>C:\Users\Nakano\source\Projects\Nakano\Myname\Debug\Myname1.exe : fatal error LNK1120: 1 件の未解決の外部参  
1>プロジェクト "Myname1.vcxproj" のビルドが終了しました -- 失敗。  
===== ビルド: 0 正常終了、1 失敗、0 更新不要、0 スキップ =====  
|
```

At the bottom of the window, there are tabs for 'エラー一覧' (Error List) and '出力' (Output). The status bar at the very bottom shows '準備完了' (Ready).

ビルド失敗

プログラムの実行については次回行う