

## Windows PC とや MAC に gcc コンパイルできる環境の構築

### 1. Windows

#### 1.1 Windows 10 の場合

Windows 10 には Ubuntu を含む Linux を実行できるサブシステム (Windows Subsystem for Linux, 以下 WSL) が導入できるので、この方法について解説する。

まず Windows 10 を最新の状態にアップデートしておく。

スタートメニュー → 設定 → 更新とセキュリティ → Windows Update ページの「更新プログラムをチェック」をクリック

更新があれば支持に従ってアップデート

つぎに以下のページを参考にして WSL をまずインストールし、その後再起動して、次にスタートメニューの中の「Microsoft Store」を起動して「Ubuntu」を選択してインストールする。

<https://www.atmarkit.co.jp/ait/articles/1608/08/news039.html>

<https://www.atmarkit.co.jp/ait/articles/1903/18/news031.html>

インストール途中でユーザ名とパスワードの設定を求められるので、入力する。

```
Enter new UNIX username: taro
```

```
Enter new UNIX password: zzzz
```

```
Retype new UNIX password: zzzz
```

インストールが成功したら、スタートメニューから Ubuntu を選ぶことで、Ubuntu を起動できる。

起動したら、以下のコマンドを入力して Ubuntu を最新の状態にする。

```
sudo apt update
```

ここでパスワードの入力を求められるので、Ubuntu をインストールしたときに入力したパスワードを入力する (Windows のパスワードではないので、注意)

```
sudo apt upgrade
```

途中で

```
Do you want to continue? [Y/n]
```

と入力を求められるので、そのまま「Enter」を押す。

また途中でインターネット接続が切断される場合、サイドコマンド入力して正常終了するまで繰り返す。

次に `gcc` を導入する。

```
sudo apt install gcc
```

これも途中で

```
Do you want to continue? [Y/n]
```

と入力を求められるので、そのまま「Enter」を押す。

以上で `gcc` を実行する環境は整っている。

なお、日本語のメッセージやオンラインマニュアルが利用したければ、以下を `Ubuntu` ウィンドウで入力しても良い（必須ではない）。

```
sudo apt -y install language-pack-ja
```

```
sudo update-locale LANG=ja_JP.UTF8
```

```
sudo apt -y install manpages-ja manpages-ja-dev
```

次に `WSL` を使ってプログラムを入力する。

プログラムを格納するフォルダをデスクトップで

右クリック → 新規作成 → フォルダ

で作成する。名前は「`programming`」とつけたとする。

このフォルダ内に `C` プログラムのソースファイルをエディタで入力し保存する。

ここではこれを `program1.c` として解説する。

`Ubuntu` を起動してこのフォルダにディレクトリを変更する。たとえば、`Ubuntu` のウィンドウで

```
cd /mnt/c/Users/xxxx/Desktop/programming
```

`xxx` は **Windows** のユーザー名なので注意。`Ubuntu` 導入時のユーザー名ではない。

ユーザー名がわからない場合は

```
ls /mnt/c/Users
```

と入力するとユーザー名一覧が表示するので、これから判断する。

またユーザー名にスペースが含まれる場合は、スペースを”¥ “と置き換えて入力する。

例 ユーザー名が `Yamadai Taro` であれば `Yamadai¥ Taro` と入力する。

`programming` フォルダに `cd` が成功したら、`Ubuntu` ウィンドウで

```
gcc -o program1.exe program1.c
```

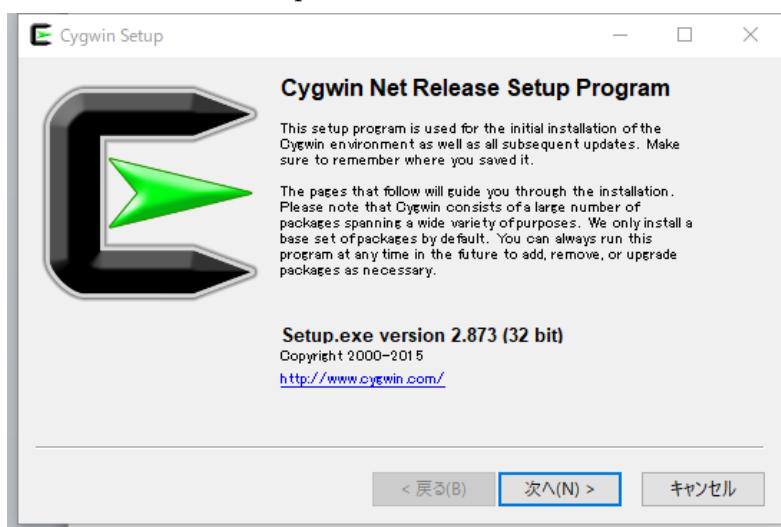
などを入力してコンパイルしてみて、デバッグする。

## 1.2 Windows 10 ではない Windows 利用の場合、あるいは 1.1 の WSL 導入手順がうまく行かない場合

Windows には unix コマンドや gcc を提供する cygwin と呼ばれるパッケージが存在する。これをインストールする方法について解説する。

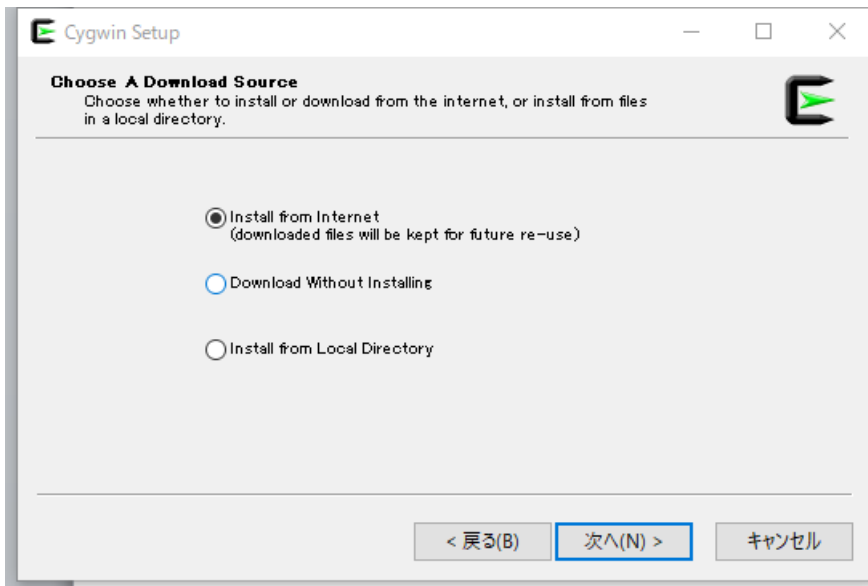
Cygwin のページの Install Cygwin から `setup-x86_64.exe` をダウンロード  
<https://cygwin.com/>

ダウンロードした `setup-x86_64.exe` をダブルクリック

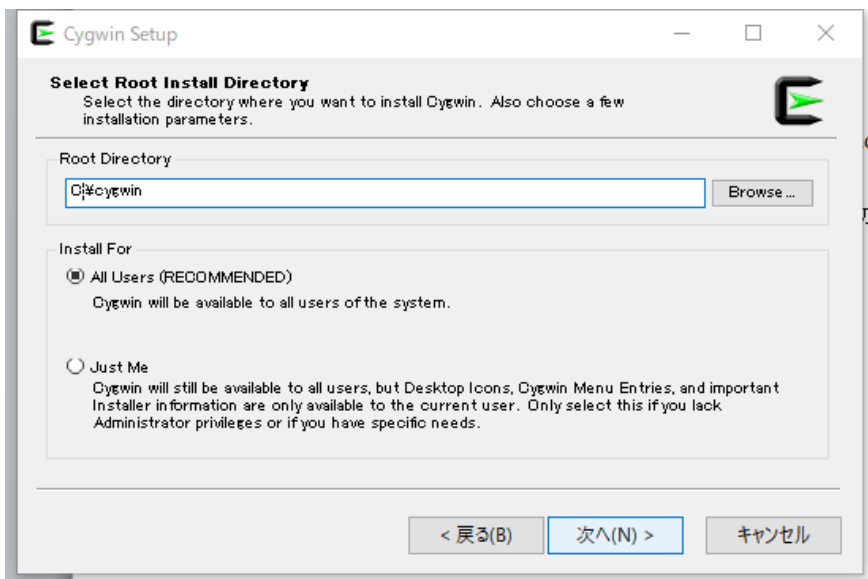


これが実行できない場合、Windows が 32 bit なので kCygwin のページの一番下にある `setup-x86.exe` をクリックしてダウンロードし、ダブルクリック

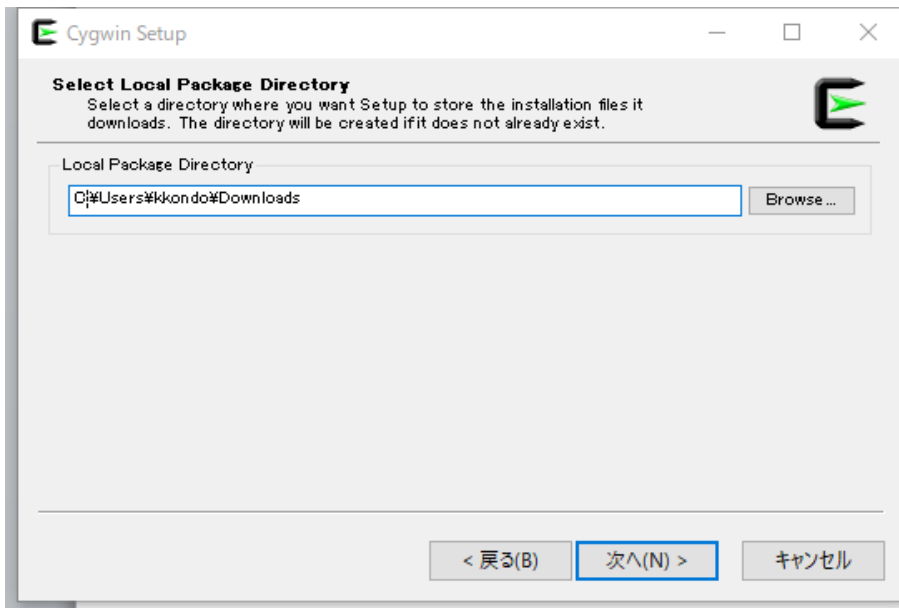
「次へ>」をクリック



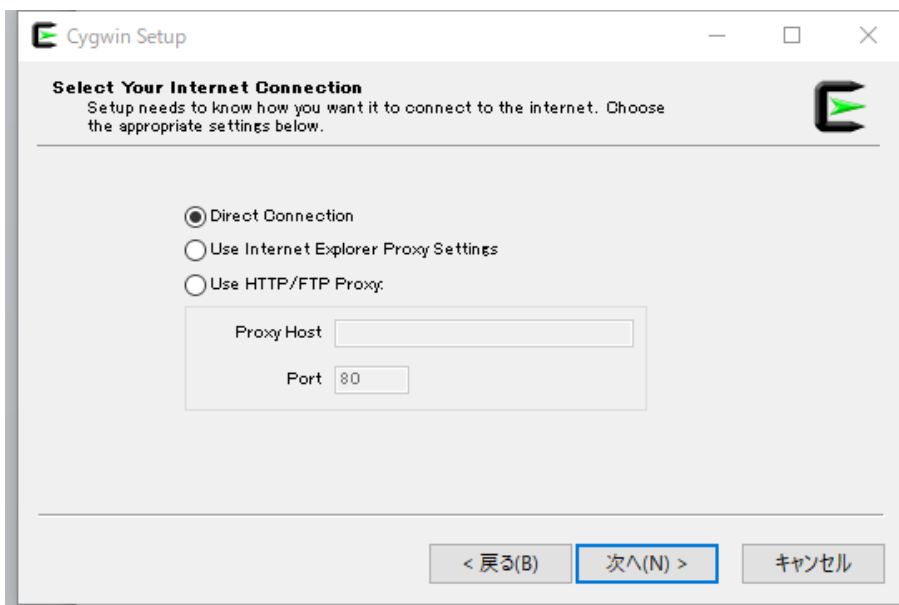
Install from Internet を選んで、「次へ>」をクリック



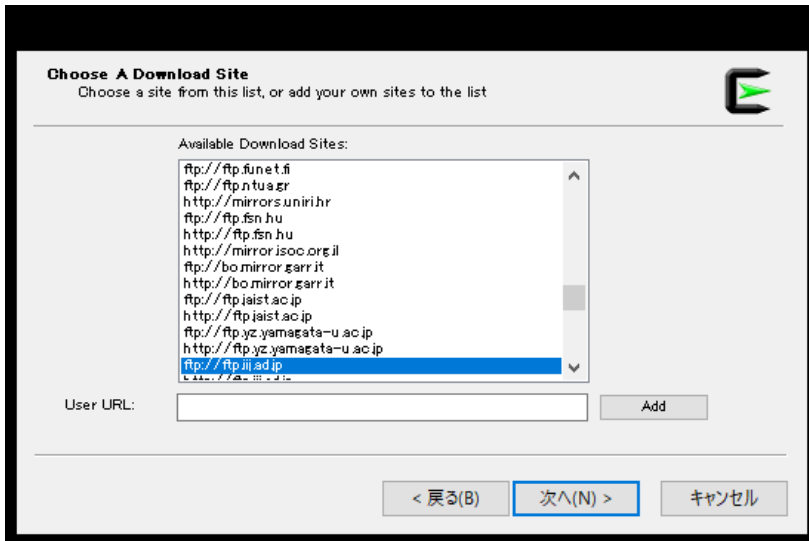
Root Directory はそのまま、  
Install For も All Users を選んで、「次へ>」をクリック



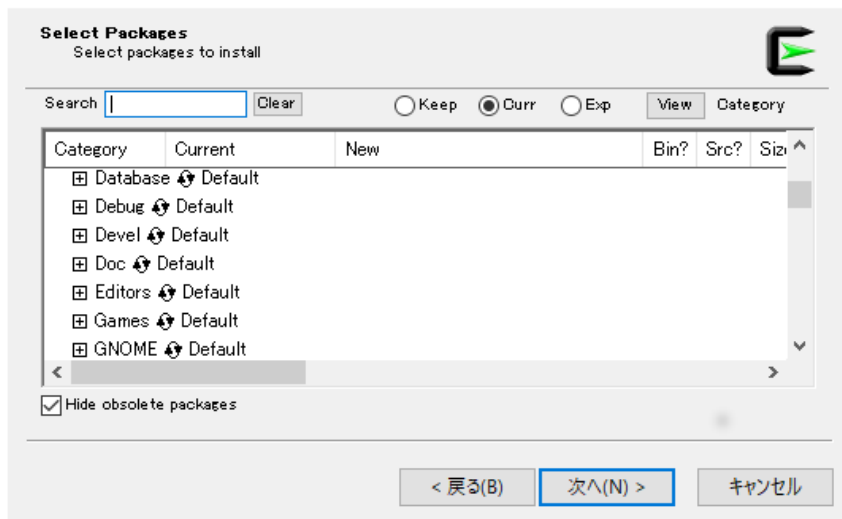
ダウンロードするデータを一時保存する場所を指定。そのままOK  
「次へ>」をクリック



インターネット接続を指定。そのまま 「次へ>」をクリック



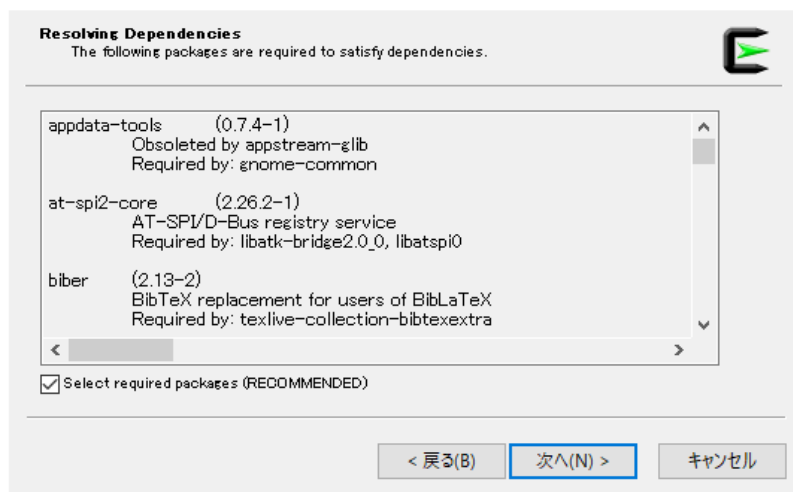
ダウンロードするサーバーを一つ以上選ぶ。おすすめや山大のサーバー  
ftp://ftp.yz.yamagata-u.ac.jp か http://ftp.yz.yamagata-u.ac.jp  
「次へ>」をクリック



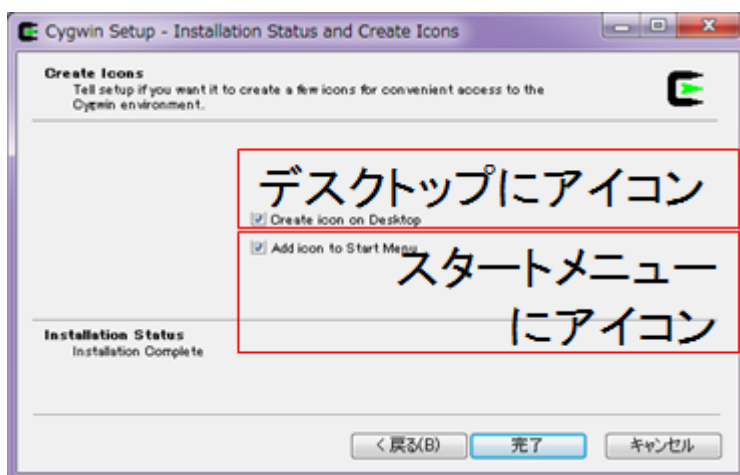
ダウンロードするパッケージを選ぶ。gccも導入したいので、Develの右のDefaultをクリックしてInstallにする

Category	Current	New
Database	Default	
Debug	Default	
Devel	Install	
Doc	Default	
Editors	Default	
Games	Default	

「次へ>」をクリック



「次へ>」をクリック



デスクトップにアイコンを置くか、スタートメニューにアイコンを加えるかは好みで。  
「完了」をクリック

デスクトップにできた Cygwin Terminal のアイコンをダブルクリックするか、  
スタートメニューの Cygwin->Cygwin Terminal を選択

端末が出てくるはず。

端末で

```
pwd [ENTER]
```

と入力すると

```
/home/xxxx
```

(xxxx はユーザ名)

と表示されるはず。これを覚えておくこと。

下記も参考にされたい。

Cygwin インストール

<http://sw.it.aoyama.ac.jp/2011/CP1/cygwin>

### 1.3 gedit のインストール

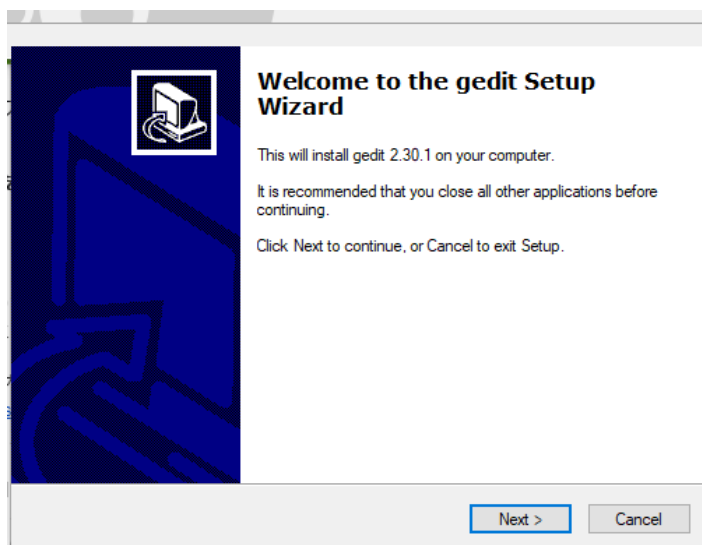
プログラムの入力は Windows に付属するアプリ「メモ帳」でも十分入力できるが、テキストエディタの gedit ウィンドウズ版も使えるので、この導入方法について解説する。

なお、Windows 版 gedit はファイル入力や保存のダイアログで反応がなくなることがあるようだが、暫く待つと反応が帰ってくることもある。

gedit-setup-2.30.1-1.exe を下記からダウンロード

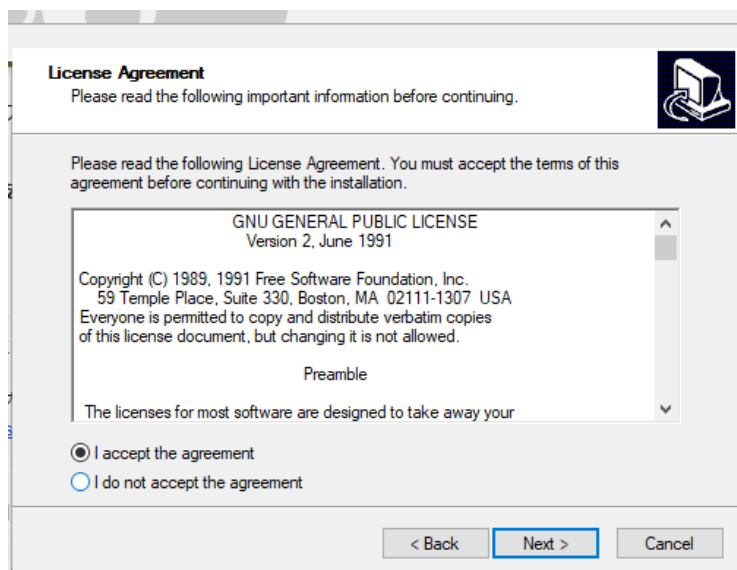
<https://ftp.gnome.org/pub/gnome/binaries/win32/gedit/2.30/>

ダウンロードしたファイルをダブルクリック

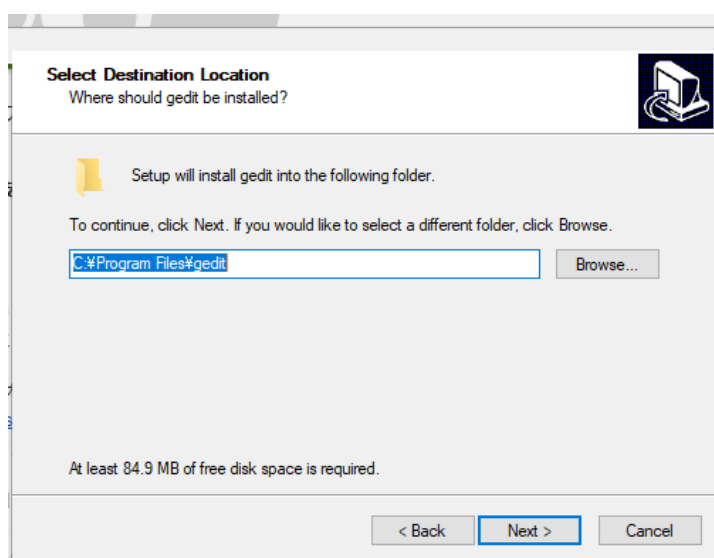




Next> をクリック



I accept the agreement を選んで、Next>をクリック



インストールするディレクトリを選んで、Next>をクリック

Select Start menu folder はそのまま Next>をクリック

Select Additional Tasks もそのまま Next>をクリック

Ready to install で Install をクリック

Completing the gedit Setup Wizard で Finish をクリック

これでインストールは完成。

スタートメニューに gedit が追加されているので、これをメニューから選ぶ

C 言語のプログラムを入力してみてよ。

セーブするときには

ファイルメニューから別名で保存を選んで、

他の場所をクリックし、下記などを選んで保存せよ。

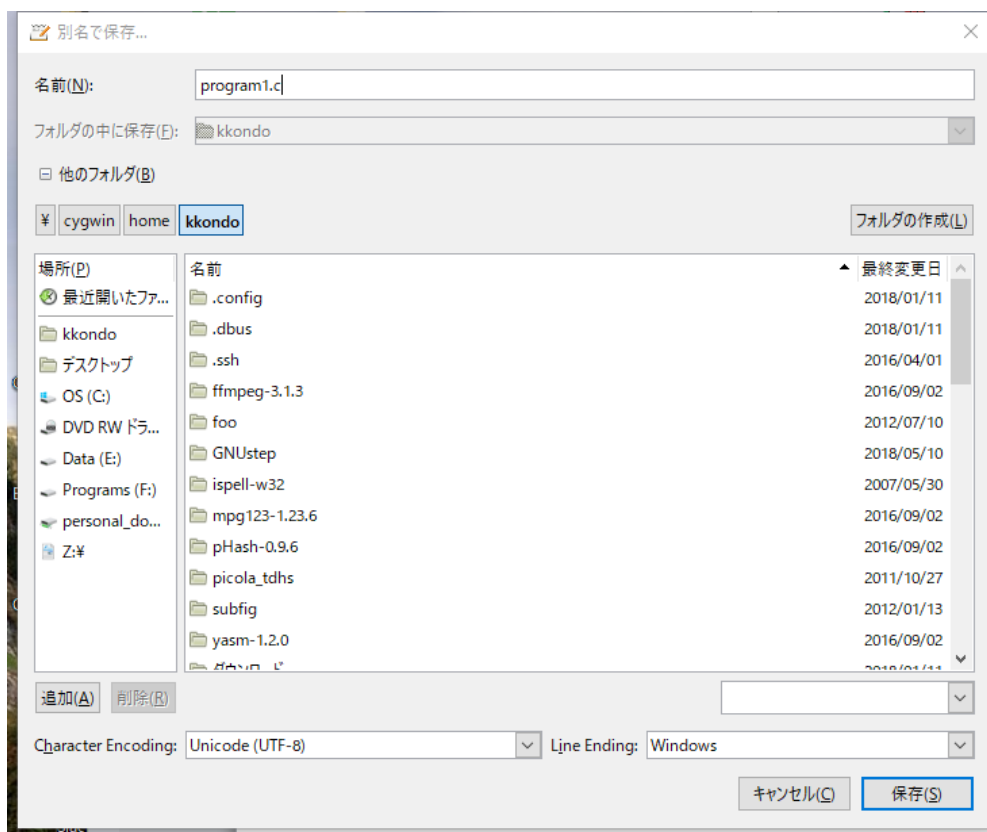
例えば Cygwin の場合、

(Cygwin をインストールしたディレクトリ)/home/xxxx

例えば

C:\cygwin\home\xxxx

ファイル名も入力



これでCのプログラムを保存して、gcc を使ってコンパイルできる。

## 2. Mac

残念ながらこちらには動作を確認する MAC がないので、Mac への導入を解説する URL を紹介するので、こちらを参考に環境構築を試みられたい。

【超初心者】Mac でプログラミング (C 言語) を始めるための準備

<http://sskdevelopment.com/programming/2018-1-17/>