

## 準備

- **Cygwin64 Terminal**や**秀丸エディタ**がデスクトップにない人は、スタートメニューから探してデスクトップにショートカットを作っておく。(デスクトップにドラッグ&ドロップ)
- **ドキュメント**の下に、この授業用のフォルダを作っておく。(例:prog)

Cygwinを起動したときのホームディレクトリ

全角文字・スペースは使わないように

この資料は、大学のコンピュータ演習室を利用する際の説明です。

自宅のパソコンでは、必ずしもこの通りに準備する必要はありません。

使用するエディタや授業用のフォルダは、各自のパソコンにより異なります。

Cygwinを起動したときのホームディレクトリは、大学のコンピュータ演習室では「ドキュメント」ですが、各自でCygwinをインストールしたときは、「C:¥cygwin64¥home¥[ユーザー名]」の下となります。

各自のパソコンに合わせて準備してください。

# Cygwinの設定

「Cygwin64 Terminal」を起動

①左上のアイコンをクリック

②[Options]を選択

③[Text]を選択

④[ja\_JP]を選択

⑤[UTF-8]を選択

⑥適当なフォントを選択

[OK]で設定完了→Cygwinを再起動して設定反映

自宅のパソコンにCygwinをインストールした人は、このようにCygwinの画面で設定を行ってください。

Cygwinのバージョンによっては、画面が違うかもしれませんが、対応する項目の設定を行ってください。

# ソースプログラムの作成

- 秀丸エディタでテキストファイルを作成
- 名前を付けて保存（授業用フォルダの下）

xxxxxxxx.cというファイル名  
全角文字・スペースは使わないように

ファイル名(N): ex01.c

ファイルの種類(T): C言語ソースファイル (\*.c)

エンコードの種類(C): Unicode(UTF-8)  BOMを付ける

改行コード(R): 自動(変更しない)

保存(S) キャンセル ヘルプ(H)

Unicode(UTF-8)を選択すること

ソースプログラムを作成する際の注意です。

パソコンでプログラムを作成するには、秀丸エディタやサクラエディタといったテキストエディタソフトを使います。

ファイルを保存する際に、エンコードの種類(サクラエディタの場合、文字コードのセット)でUTF-8を選んでください。

# コンパイル

ここからは  
Cygwinの画面

- progという授業用フォルダに移動(最初だけ)

```
$ cd prog ↵
```

- ex01.cをコンパイルしてa.exeという実行ファイルを作成

```
$ gcc ex01.c ↵
```

- ex01.cをコンパイルしてex01.exeという実行ファイルを作成

```
$ gcc -o ex01 ex01.c ↵
```

ここからはCygwinを立ち上げた画面での操作です。

前もって授業用のフォルダ(この場合prog)を作成していれば、そのフォルダに移動するために「cd prog」と入力します。

授業用のフォルダを作成していない場合は、この操作は飛ばしてください。

Cygwinでのプログラムの作成手順は、ソースファイルをコンパイルして実行ファイルを作成します。

MacやLinuxの場合も同じですが、実行ファイルがa.exeでなく、a.outとなります。

なお、paiza.ioなどWebからプログラムを作成する場合や、モバイルCを使う場合は、コンパイルする必要はありません。

## 実行

- a.exe を実行

```
$ ./a ↵
```

- ex01.exe を実行

```
$ ./ex01 ↵
```

実行ファイルを実行する際に、先頭に「./」をつけるのを忘れないでください。

「.exe」は省略可能です。

MacやLinuxでは、「./a.out」と実行してください。


# デバッグ

- コンパイルでエラー

```
$ gcc ex01.c
ex01.c: 関数 'main' 内:
ex01.c:26:5: エラー: expected ';' before 'return'
return 0;
    ^
```

- 実行結果がおかしい

```
$ ./a
整数型 : 1
実数型 : 7182133658216492664008893980735044874213231
000000000000000000000000000000000000000000000000000000000000000000000000
```

 ソースプログラムを修正してコンパイル、  
実行を繰り返す

プログラムを作成して、それがそのまま問題なく動作することはほとんどありません。

必ずどこかにプログラム上のミス(バグ)があります。

バグは、コンパイル時にわかることもありますが、実行した結果を見て初めてわかるものもあります。

バグの修正を繰り返しながらプログラムを完成させることがプログラミングなのです。