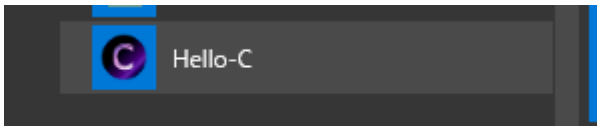
Hello C 使い方

Ver.2.0.1.4
2018.04.04 update

○基本操作

1．Hello C の起動

1）[スタート]→アプリ一覧から[Hello-C]をクリックすると起動します。



2）MinGW についてのメッセージボックスが出た場合

→　エラー文を日本語化したい場合は[はい]

→　何もしない場合は[いいえ]　※デフォルトで英語になります。

※「(ドライブ名):¥」へ MinGW フォルダ、及びファイルを追加します。既に存在する場合上書き追加します。既存の変更に注意してください。

2．Hello C の終了

1）メニューバーの[ファイル]をクリックします。

2）[終了]をクリックします。

3）プロジェクトの保存が必要な場合は通知されます。
必要な操作を選択してください。

○プロジェクト

ソースや実行ファイルはプロジェクトごとに管理されます。

1．プロジェクトの作成

1）メニューバーの[ファイル(F)]をクリックします。

2）[新しいプロジェクト]をクリックします。

3）作成したいプロジェクト名を入力します。

4）プロジェクトが作成され、コードを打ち込めるようになります。

2．プロジェクトを開く

1）メニューバーの[ファイル(F)]をクリックします。

2）[プロジェクト一覧]をクリックします。

3）開きたいプロジェクトを選択し、ダブルクリックします。
※選択した状態で右クリック→[開く]でも開くことができます。

3．プロジェクトを閉じる

1）メニューバーの[ファイル(F)]をクリックします。

2）[プロジェクトを閉じる]をクリックします。

4．プロジェクトの名前を変更する

1）メニューバーの[ファイル(F)]をクリックします。

2）[プロジェクト一覧]をクリックします。

3）名前を変更したいプロジェクトを選択します。

4）右クリック→[名前の変更]をクリックします。

5）新しい名前を入力して[OK]を押します。

※プロジェクトは閉じておく必要があります。

5．プロジェクトを削除する

1）メニューバーの[ファイル(F)]をクリックします。

2）[プロジェクト一覧]をクリックします。

3）削除したいプロジェクトを選択します。

4）右クリック→[削除]をクリックします。

5）[はい]をクリックします。

※プロジェクトは閉じておく必要があります。

〇ソースファイル

ソースファイルの一覧は「ファイルビュー」から見ることができます。

ファイルビューを表示するには、メニューバーの[表示]から[ファイルビュー]を選択してください。

1．ソースファイルを追加する

1）メニューバーの[ファイル(F)]をクリックします。

2）[追加]から、追加したいファイルの種類を選択します。

3）ファイル名を入力します。

2．ソースファイルの名前を変更する

1）ファイルビューを表示しておきます。(前述)

2）名前を変更したいファイルを選択します。

3）右クリック→[名前の変更]をクリックします。

4）新しい名前を入力して[OK]を押します。

3．ソースファイルを削除する

1）ファイルビューを表示しておきます。(前述)

２）削除したいファイルを選択します。
３）右クリック→[削除]をクリックします。
４）[はい]を押します。
※ファイルがすべて削除された場合でもプロジェクトは管理されます。

〇実行

作成したコードを実行したり、バグを見つけるデバッグを行ったりすることができます。

１．プログラムを実行する
１）実行したいプロジェクトを開いておきます。
２）ツールバーの[Debug]のチェックを外します。
３）ツールバーから[実行]を押します。
４）コンパイルに成功した場合はプログラムが実行されます。
失敗した場合はエラーや警告が「エラー一覧」に表示されます。

※「エラー一覧」を表示するには、メニューバー[表示]から[エラー一覧]を選択してください。
※「コンパイルに成功したが実行ファイルが見つからない」と出た場合は、エラー一覧から警告内容を修正し再度実行してください。なお、既に実行ファイルが存在する場合は、前回のコンパイル段階での実行ファイルが実行されます。（現バージョンではこの通知はありません）

２．デバッグ機能を使用する

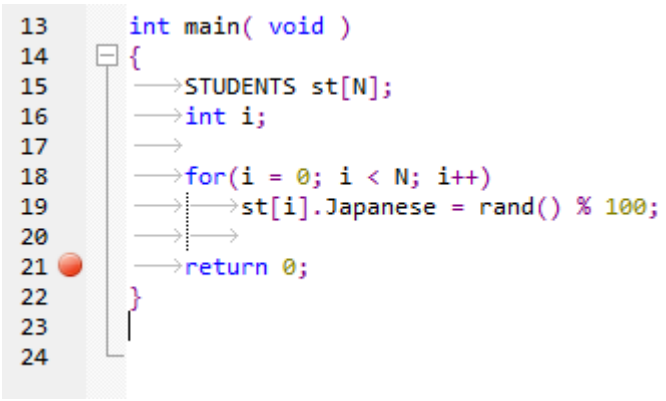
A）デバッグの開始と終了

１）実行したいプロジェクトを開いておきます。
２）ツールバーの[Debug]にチェックをします。
３）ツールバーから[実行]を押します。
４）コンパイルに成功した場合はデバッグが開始されます。
失敗した場合はエラーや警告が「エラー一覧」に表示されます。
コードを修正し再度実行してください。
※「エラー一覧」を表示するには、メニューバー[表示]から[エラー一覧]を
選択してください。
５）プログラムが停止する原因があった場合は通知されます。
→　プログラムを続行するには[続行]
→　プログラムを強制終了するには[強制終了]
→　プログラムの停止位置を確認したい場合は[閉じる]
６）プログラムが終了した時点でデバッグは終了されます。
途中で終了したい場合は、ツールバーの[■（停止）]をクリックしてください。

B）ブレイクポイントの設定・解除

ブレイクポイントを設定することで、プログラムを設定場所で一時停止させ、変数の中身を確認することができます。

1）デバッグしたいプロジェクトを開いておきます。
2）行番号の右側をクリックすることでブレイクポイントを設定・解除できます。



3）ブレイクポイントで設定された行でプログラムが止まります。
4）ブレイクポイントで停止後、その時点で表示できる変数が「デバッグ・変数」に表示されます。見方については後述をご覧ください。
※「デバッグ・変数」を表示するには、ツールバー[表示]から[デバッグ・変数]を選択してください。

C）ステップイン、ステップアウト、ステップオーバー

これらは、ブレイクポイントで停止後、1行ずつ実行したい場合に使用します。

ステップイン：
現在のステートメントを実行し次のステートメントで停止します。
関数である場合は関数の内部へ移動します。

ステップアウト：
現在のステートメントを実行し次のステートメントで停止します。
関数である場合は、その関数全体を実行したのち、呼び出し後の次のステートメントで停止します。

ステップオーバー：
現在実行している関数全体を実行したのち、呼び出し元へ移動します。

1）実行したいプロジェクトを開いておきます。
2）ブレイクポイントを設定します。
3）デバッグを開始します。
4）ブレイクポイントで停止した時点でこれらの操作が有効になります。
5）ツールバーの各アイコンをクリックしてください。

左から、[ステップイン]、[ステップアウト]、[ステップオーバー]になります。
効果は前述の通りです。

D）デバッグ・変数の見方

ブレイクポイントやステップでプログラムが一時停止した場合、自動でその時点で表示可能な変数の中身が表示されます。
※「デバッグ・変数」を表示するには、ツールバーの[表示]から[デバッグ・変数]を選択してください。

- 変数名、値、アドレスの順で表示されます。
- 配列、構造体は階層構造で表示されます。
  中身を展開したい場合は、「+」ボタンをクリックしてください。
- 表示されるたびにすべて展開しておきたい場合は、ウィンドウ左上の「すべて展開」にチェックしてください。
- 前回から値が変更された場合は背景色が変わり、「以前の値→現在の値」が表示されます。

・指定したメモリのみ表示する場合

1．表示したいメモリの一番右にあるチェックボックスをチェックします。
※アドレスが表示されているメモリに限ります。
2．上にある[選択表示]ボタンをクリックすることで選択されたメモリのみ表示されます。
3．再度全体を表示するには[全体表示]ボタンをクリックしてください。

〇オプション

オプションでは Hello C の設定が行えます。
オプションを表示するには、ツールバーの[ツール]から[オプション]を選択してください。

- 全般
  ・配色テーマ
    Hello C の全体の色を決めることができます。明色は白、暗色は灰色基調になります。
  ・プロジェクトの場所
    プロジェクトを管理するフォルダを指定します。
    デフォルトはドキュメントです。
- 起動時
  ・最大化する
    ウィンドウを最大化します。
  ・前回のレイアウト配置で開く
    ソフトを再起動しても、前回のレイアウト配置で起動します。
  ・MinGW 日本語化についてのメッセージを出す
    MinGW の設定について、再度問い合わせるようにします。
- バージョン情報
  現在の Hello C のバージョンを確認することができます。

〇エディタ設定

エディタ設定ではエディタに関する設定が行えます。
エディタ設定を表示するには、ツールバーの[ツール]から[エディタ設定]を選択してください。

- 全般
  ・ユーザー補助
    ユーザーの補助に関する設定を変更できます。
  ・環境
    ・外部によるファイル内容変更時、自動再読み込みを行う
      ソースファイル、テキストファイル等が外部変更、または実行されたプログラムによる変更時、自動で再読み込みを行うかを設定できます。
      手動で再読み込みを行う場合は、ファイルビューから操作してください。
- フォント
  フォントの種類とサイズを変更できます。
- テーマと色
  配色テーマではエディタの基調となる色を設定できます。
  各文字色はテーマ別に設定できます。上記の配色テーマから変更したいテーマを選び、左側の欄

からさらに変更したい種類を選択し、[変更]をクリックしてください。

〇コンパイル設定

- 全般

・コンパイル時に構文チェックを行う

コンパイル時に構文チェックを行うかを設定します。

・実行時に”pause”コマンドを適用する

プログラム終了時にウィンドウが閉じられないようにします。

※デバッグ時は適用されません。

・続行確認文字を表示しない

デフォルトで「続行するには何かキーを押してください...」と出ますが、これを表示しません。

・-mwindows オプションを使用する

WinAPI を使用する際に設定します。

- コマンドライン引数

コマンドライン引数を設定できます。

新しくプロジェクトを開いた場合にこの設定を無効化するには[プロジェクトを開いた際に無効にする]にチェックしてください。さもなければ、別のプロジェクトで設定した引数がそのまま適用されることになります。