

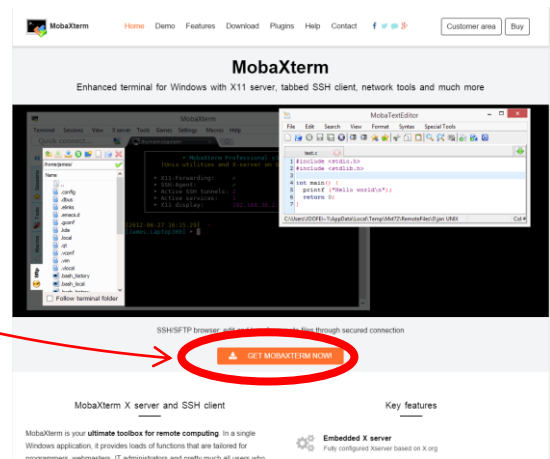
自宅の Windows でのプログラミング環境の構築

● MobaXterm を使用したプログラミング環境の構築

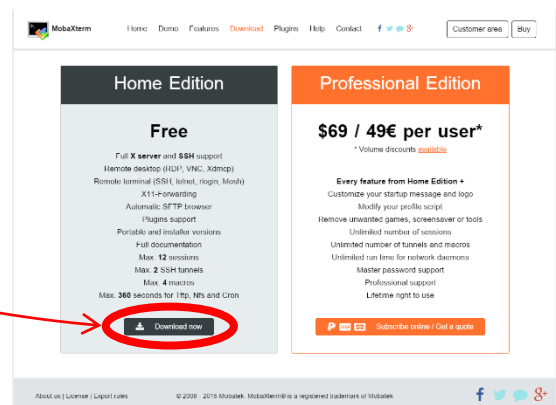
① MobaXterm のダウンロード&インストール

以下の手順でサイト (<http://mobaxterm.mobatek.net/>) からダウンロードする。

「GET MOBAXTERM NOW!」をクリック

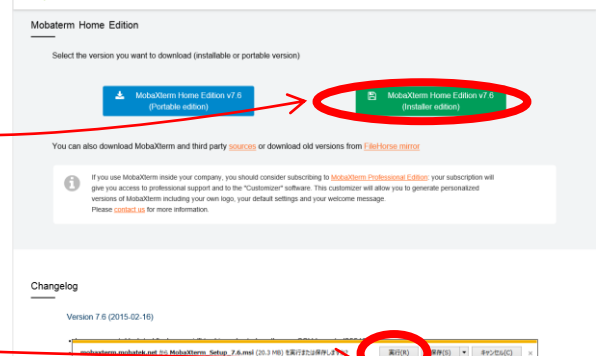


「Download now」をクリック



「MobaXterm Home Edition vX.X (Installer edition)」

をクリック

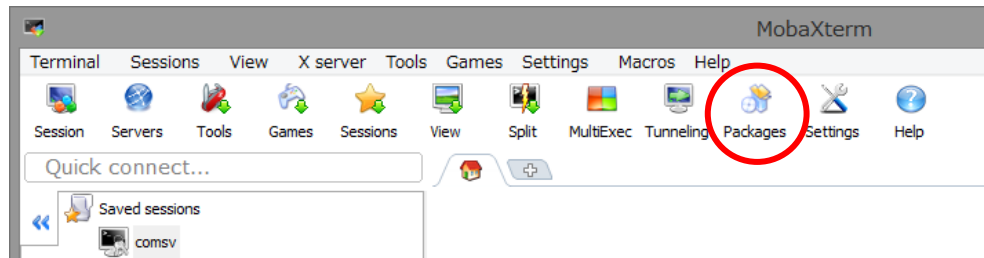


「実行」をクリック

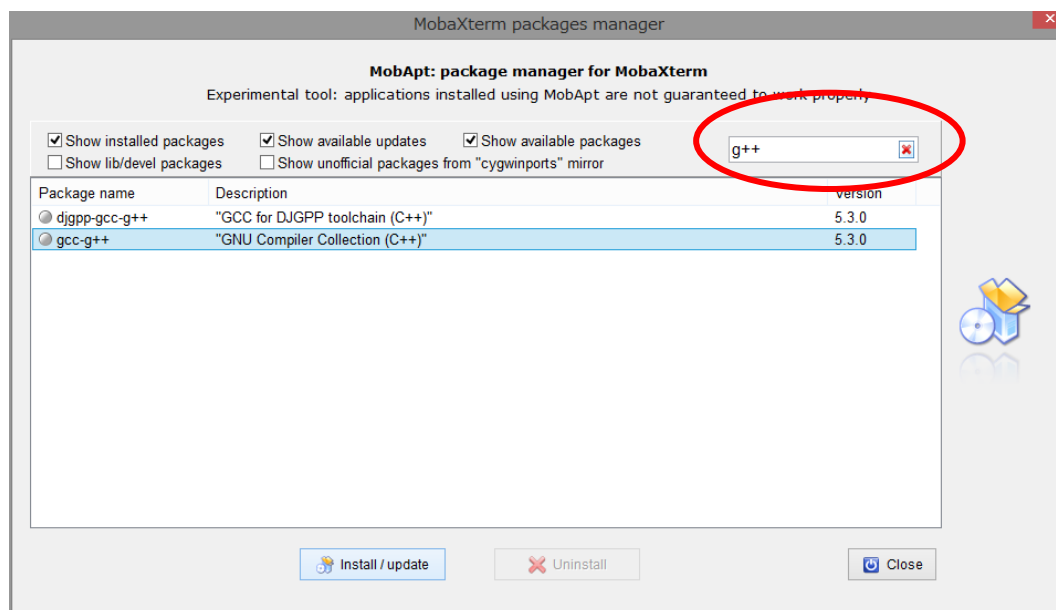
→ ダウンロードした zip ファイルを解凍した後
MobaXterm_installer_XXX をクリックし、インストール(XXX はバージョン番号)
(zip ファイルの解凍方法は、zip ファイルを右クリックして「すべて展開」を選択)

② C/C++コンパイラをダウンロード&インストール

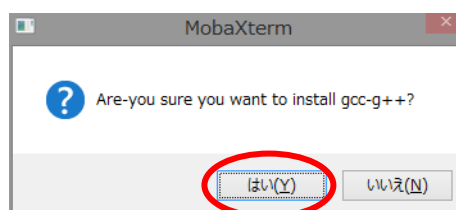
MobaXterm を起動し、以下のような画面の「Packages」をクリックしてください。ここから「Trying to download file…」などが出てきてしばらく待たされると思います。



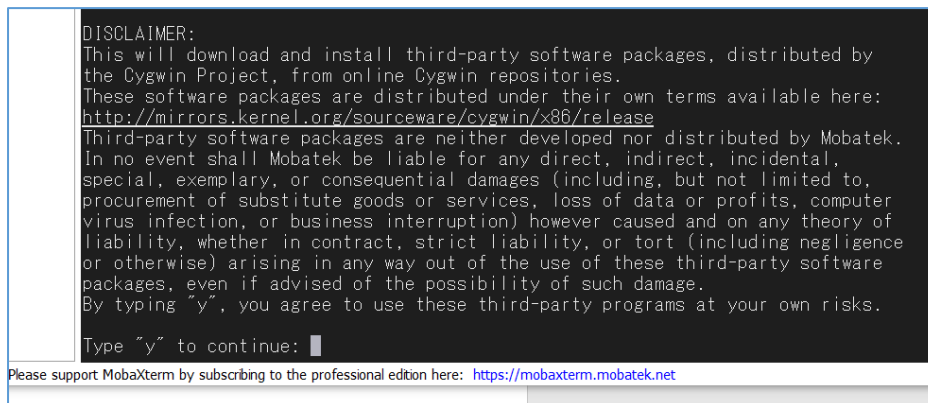
その後に出てきた以下のような画面の赤丸で示す「<Filter packages>」のところに「g++」と入力すると、以下のように選択肢が絞られると思います。そして、「gcc-g++」を選択し、下の「Install/update」ボタンを押してください。



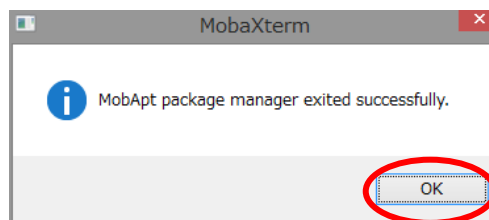
次に出てくる以下の画面は「はい」を選択してください。すると、またしばらく待たされます。



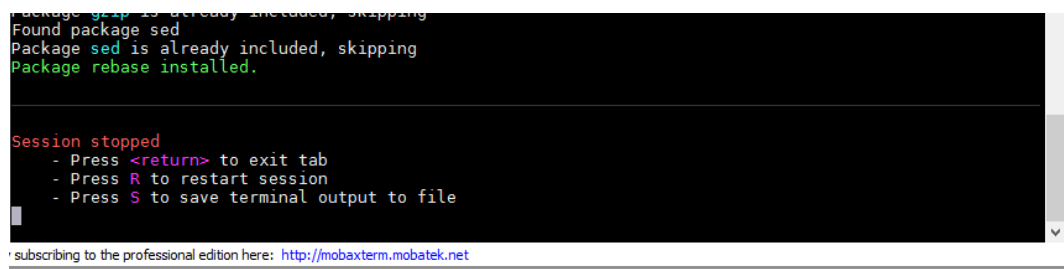
この画面がでてきたら、y を押して下さい。



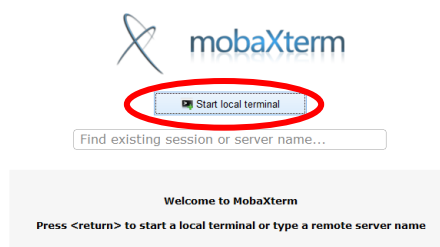
すべてのインストールが自動で行われ、最後に以下の画面が出てきますので、「OK」を押します。



以下のような画面になるので、「Enter キー」を押してください。

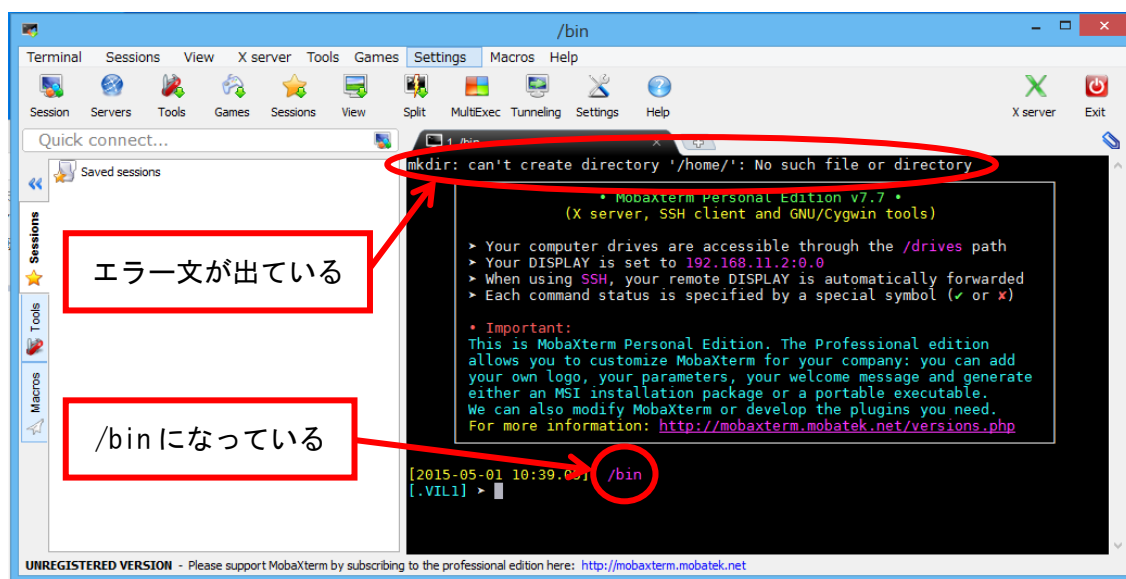


以下の中央の「Start local terminal」をクリックしてください。これで準備完了です。



※パソコンのアカウント名に日本語の名前を使用している場合

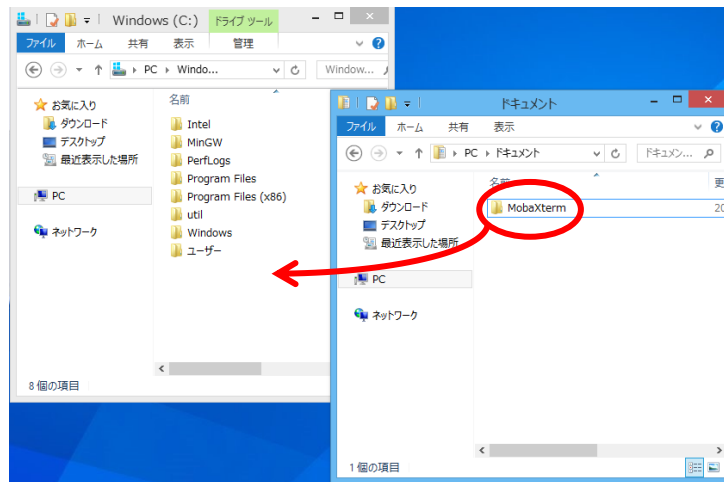
自身のアカウントに日本語の名前を使用すると、MobaXterm を正しく起動することができません。具体的には、MobaXterm を起動したときに以下のような画面になっているとき、正しく起動できていないことを表しています。



これを解決するための手順を以下に示します。

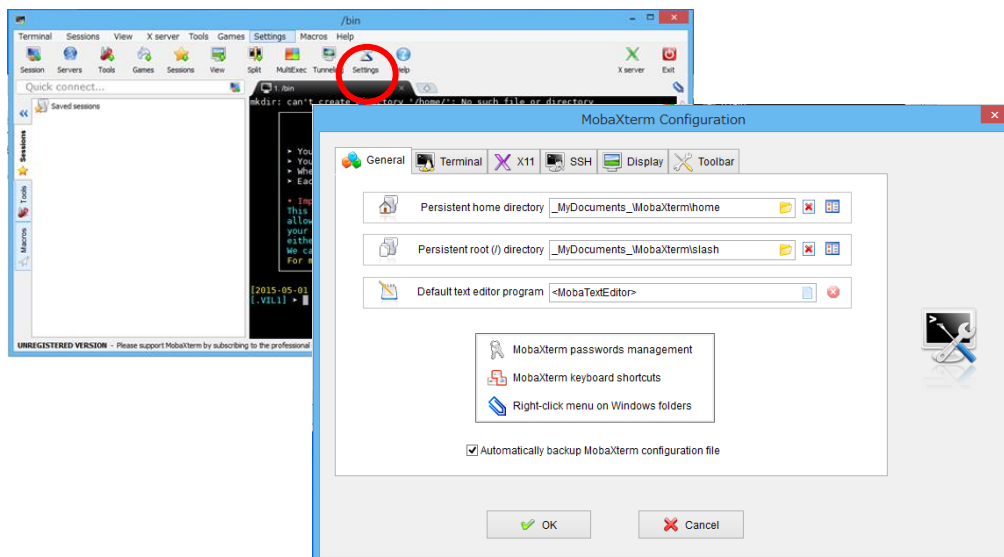
① MobaXterm のフォルダの移動

ドキュメント (Document) フォルダにある MobaXterm を C ドライブ直下 (コンピューター→ローカルディスク(C:) [Program Files フォルダがある場所]) にコピーします。



② 設定の変更

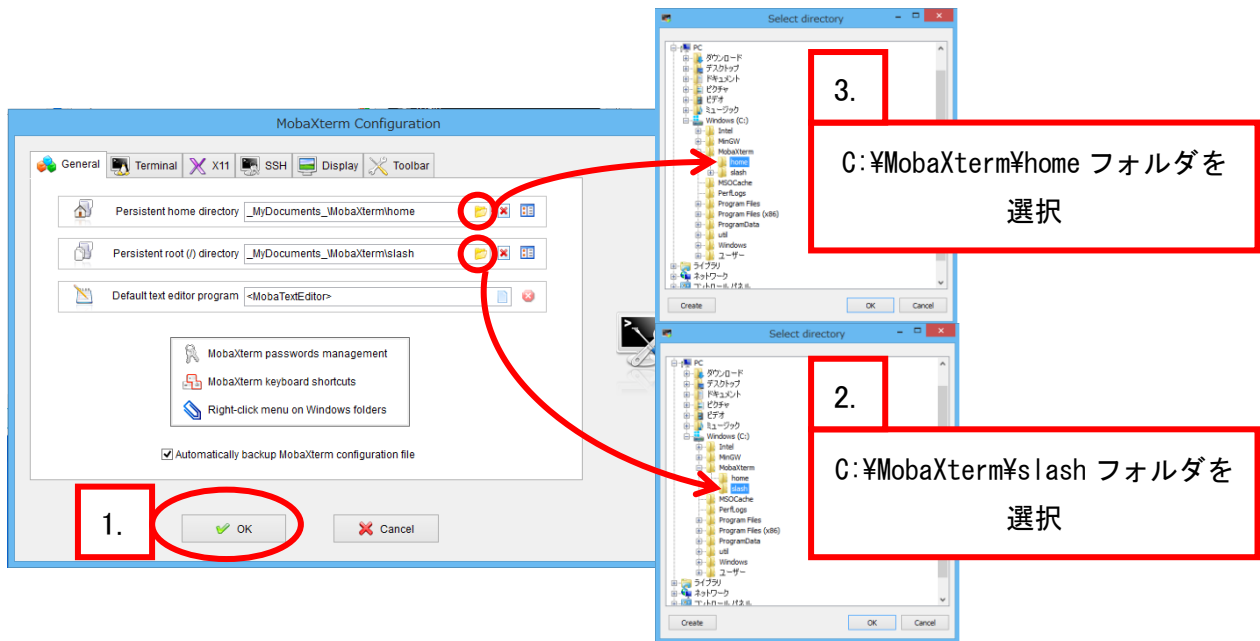
MobaXterm を起動し、「Settings」を選択して設定画面を出します。



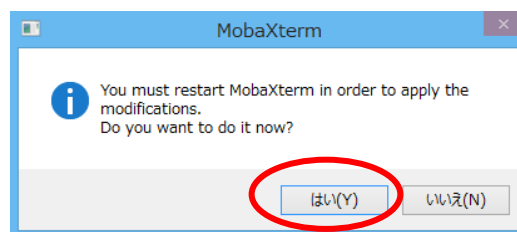
次に、Persistent home directory と Persistent root(/) directory の場所を変更します。

変更後の場所は、手順①でコピーした先(Cドライブ直下の MobaXterm)のフォルダを以下

のように選択します。それができたら「OK」ボタンをクリックしてください。



「OK」ボタンをクリックと、以下のように MobaXterm の再起動を促すウィンドが出てくるので、「はい」をクリックします。MobaXterm が再起動されたら、エラー文等が出ていないことを確認してください。



● MobaXterm の使用方法

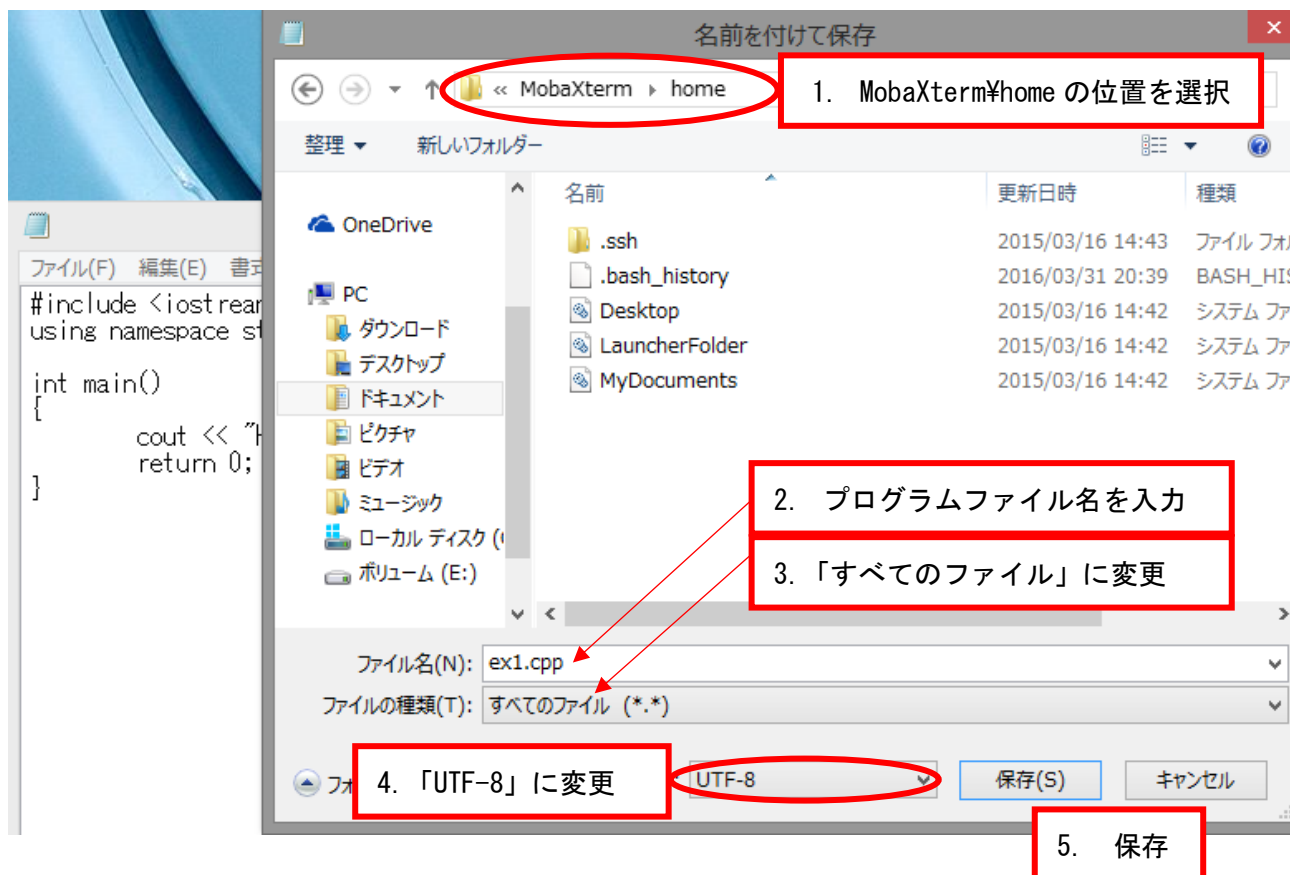
① ホームディレクトリ（デフォルトのフォルダ）の確認

MobaXterm を起動すると、最初のディレクトリは「ドキュメント¥MobaXterm¥home¥」になっている※。つまり、デフォルトの状態では、この「home」フォルダの中にプログラムがあれば作成したプログラムがコンパイルできる。

※パソコンのアカウント名に日本語の名前を使用している場合は「C:¥MobaXterm¥home¥」になっている

② プログラムの作成

Windows には標準で emacs がないので、代わりにメモ帳（notepad）を利用する。メモ帳を開いてプログラムを書き、上記の home ディレクトリに保存する。



③ プログラムのコンパイル

大学の環境と同様に、以下のコマンドでコンパイルする。

```
g++ XXXX.cpp -o XXXX
```

④ プログラムの実行

大学の環境と同様に実行プログラムを実行すると、何も出力されない。以下は、「Hello World!」と出力する実行プログラム「test」を実行した結果である。

```
[2015-03-31 17:29.23] ~  
[NaoyukiAwano.awano] > test
```

大学の環境とは異なり、実行ファイル名の前に「./」（ドット スラッシュ）をつけないとプログラムが起動しない。以下は「./」を付けて実行した結果である。

```
[2015-03-31 17:29.40] ~  
[NaoyukiAwano.awano] > ./test  
Hello World!
```


- おまけ1（その他のテキストエディタ）

メモ帳以外にもテキストエディタはある。代表的なテキストエディタの例を以下に示しておく。

メモ帳の場合は文字に色がつかないが、以下の中には色を付けてくれるものもある。

サクラエディタ : <http://sakura-editor.sourceforge.net/>

Sublime Text 3 : <http://www.sublimetext.com/3>

TeraPad : <http://www.forest.impress.co.jp/library/software/terapad/>

K2Editor : <http://k2top.jpn.org/?K2Editor>

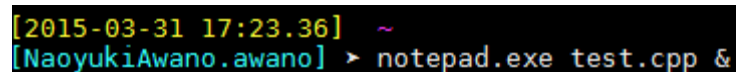
- おまけ2（手順②の簡単化）

メモ帳の場合、手順②を以下のようなコマンドでも開くことができる。

notepad.exe XXXXX.cpp &

例えば、「test.cpp」というソースファイルを作成する場合は、以下のようなコマンドを実行す

る。すると、メモ帳が開くので、任意のプログラムを書き込む。



```
[2015-03-31 17:23.36] ~  
[NaoyukiAwano.awano] > notepad.exe test.cpp &
```