*Nota: per il momento non esiste una soluzione universale al problema di linking di Postgre con Windows; ho raccolto tutto quello che ho trovato. Gabriel*

Fase (1): Avere un compilatore C/C++ installato.

Si può ottenere in vari modi (*uno alternativo all’altro*):

1. Scaricare MSYS2 da: <https://msys2.github.io/> e seguirne le istruzioni

Una volta che l’installazione è completata, nella “MSYS2 Shell” si seleziona un’opzione tra MSYS2 64-bit o MSYS2 32-bit (oppure *mingw64.exe* o *mingw32.exe* nella cartella di installazione). Se fatto correttamente, il prompt della shell di MSYS2 segnala *MINGW64 o MINGW32*.

Si seguono poi i passaggi indicati correttamente da:

<https://stackoverflow.com/questions/30069830/how-to-install-mingw-w64-and-msys2>

1. Scaricare l’installer ufficiale dal sito:

oppure seguendo il sito: <https://nuwen.net/mingw.html> e seguendo le istruzioni fornite dalla pagina, scompattando tutto in una cartella preferita e impostando anche qui le variabili d’ambiente in modo del tutto analogo a sopra

1. Scaricare una cartella tra una zip per x32 o x64 da <https://winlibs.com/> e poi basta fare come prima, idealmente:

Scompattare in “C:\” dando un nome (es. Openlibs) e poi si mette nelle variabili d’ambiente:

*“C:\Openlibs\mingw64\bin”*

1. Fare come indica il pdf di laboratorio “*Accesso a Postgre da software”*:

Scaricando l’installer da <http://win-builds.org/doku.php>, ed il file .exe corrispondente alla propria versione di Windows. Successivamente si aprirà un’altra finestra nella quale occorrerà selezionare alcune opzioni relative al proprio dispositivo e poi la cartella in cui installare il compilatore. Selezioniamo “native windows” come sistema e “86 64” per quanto riguarda l’architettura. Cliccando poi il tasto *Browse* possiamo selezionare un’apposita cartella, che dovrà essere vuota, all’interno della quale effettuare l’installazione. La cartella vuota va creata prima di iniziare l’installazione. Nel nostro caso la cartella è “C:\MinGW64”. Su ciascuno dei pacchetti, dovrà essere selezionata l’opzione “Install”.

Fatta una tra queste, si deve impostare la variabile d’ambiente.

Ciò può essere fatto in vari modi:

* Cercando nello Start “Variabili d’ambiente”
* Cliccando su Esplora Risorse – Questo PC – Tasto destro e si clicca su Proprietà – Impostazioni di sistema avanzate.

Successivamente, sul primo e secondo pannello, quindi “Variabili d’Ambiente per Utente” e “Variabili di sistema”, si scorre finché si legge Path o PATH e si clicca modifica, aggiungendo con “Nuovo” il percorso della cartella utile, usando la cartella bin.

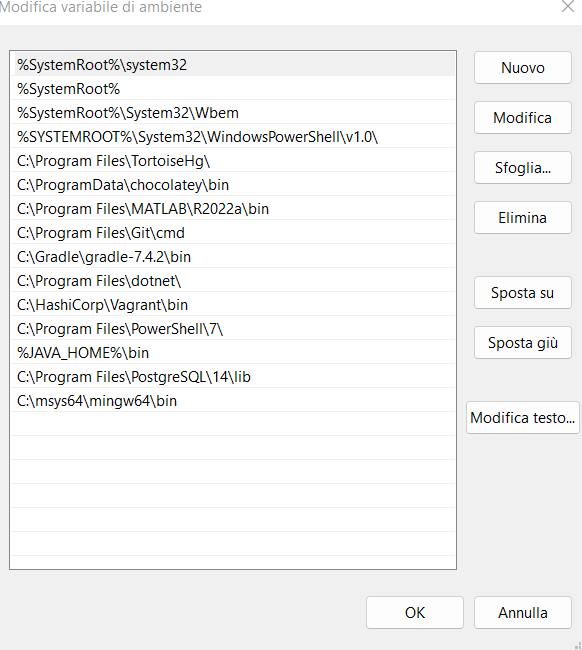
Immagine che contiene testo

Descrizione generata automaticamente

*Nota: Dovrebbe andare bene solo per Utente; io ho risolto mettendo anche in Sistema, tenendo quindi le variabili d’ambiente per Postgre (pezzo successivo) e MinGW in entrambi.*

Dallo screen, si clicchi su PATH/Path e si clicchi su Modifica e poi si clicca su “Nuovo”, aggiungendo un percorso come:

“C:\MinGW64” (prossimo screen indica un esempio):



Quindi, seguendo il lab 6, sarebbe:

*C:\Mingw64\bin*

oppure come ho fatto io scaricato MSys funziona con:

*C:\msys64\mingw64\bin*

Lo screen riguarda il pannello “Variabili utente di sistema”

Fase (2): Impostare Postgre nelle variabili d’ambiente

Sempre in PATH per Variabili di sistema si imposta la cartella lib della propria installazione di Postgre, nel mio caso:

*C:\Program Files\PostgreSQL\14\lib*

Nota:

Ad alcuni dà come errore:

*Fatal error: dependencies / include /libpq -fe.h: No such file or directory*

Ho visto anche che "libpq-fe.h" sta nella cartella *include* e non *lib* come segnala il lab.

Quindi prova a mettere nelle variabili d'ambiente:

*C:\Program Files\PostgreSQL\14\include*

O eventualmente:

*C:\Program Files\PostgreSQL\14\bin* (alcuni hanno risolto tutto mettendo bin, ad esempio)

Per eseguire il file, si ha bisogno, nella stessa cartella del file, di una cartella *dependencies* che contiene *lib* ed *include* (per comodità da me chiamato *main.cpp*)

Visione della cartella (una generica)

<C:/User/Desktop/folder>

*main.cpp*

*dependencies*

*- include (libpq-fe.h - pg\_config.ext.h - postgres\_ext.h)*

*- lib (libpq.dll - libpq.lib)*

NB: I file suddetti possono essere recuperati da *lib* ed *include* disponibili nel percorso indicato sopra per Postgre. Comunque, facendo le cose per bene, non dovrebbe neppure essere necessario.

Fase (3):

Esecuzione del file main.cpp (avendo cura di reimpostare nome utente, nome database, ecc. nel file .cpp)

*g++ main.cpp -L dependencies\lib -lpq -o main -std=c++11*

(oppure se non dovesse dare risultati l’esecuzione):

*g++ -static -libstdc++ main.cpp -L dependencies\lib - lpq -o main*

Per eseguire il file, da cmd basta scrivere ".\main",

mentre su Windows Terminal/Powershell si clicchi direttamente *l'exe* nella stessa cartella di *main.*cpp per eseguire il codice.

*Nota aggiuntiva* (non dovrebbe essere necessaria, ma è successo a diversi)

Se mancano dei driver in compilazione tipo questi (vanno tutti nella cartella lib di Postgre; se non sono presenti, scaricarli da Google e piazzarli dentro la cartella, come si vede sotto):

*libcrypto-1\_1-x64.dll*

*libintl-9.dll (potrebbe non esserci e scaricarlo da:* [*https://www.dll-files.com/libintl-9.dll.html*](https://www.dll-files.com/libintl-9.dll.html)*)*

*libssl-1\_1-x64.dll*

*libiconv-2.dll (scaricarlo da* [*https://www.dll-files.com/libiconv2.dll.html*](https://www.dll-files.com/libiconv2.dll.html)*)*

Link di OpenSSL: <https://slproweb.com/products/Win32OpenSSL.html>

(folder di riferimento: C:\Program Files\OpenSSL-Win64)

(Va bene in fase di installazione, mettere i driver richiesti nella stessa cartella di OpenSSL usando l’installer ma anche nella cartella *system32*; va bene scaricare anche il light e per comodità, bene anche l’ultima versione, quindi 3.0.3).

Se anche così non trovasse i driver (a seconda che si trovino in System32 oppure nel percorso di OpenSSL; copiarsi i driver da uno dei due percorsi (scelto in fase di installazione di OpenSSL e metterli dentro la cartella *lib* di PostgreSQL).

Oppure: nel caso possa servire, installarsi la versione più recente dei redistributable di Visual Studio, spesso fornitori di vari driver mancanti, da:

<https://docs.microsoft.com/it-IT/cpp/windows/latest-supported-vc-redist?view=msvc-170>