**FLTK installation with Visual Studio 2015**

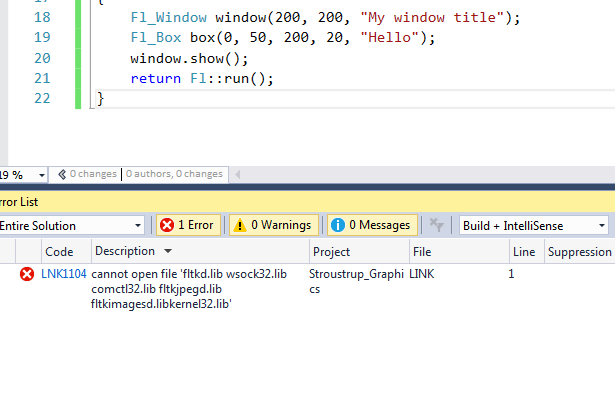
***!!! I used fltk-1.1.10 version.***

**Do Everything as described in Stroustrup Programming: Principles and Practice Using (PPP) book, Dodatek D. But there are couple things that need to be mentioned.**

1. **fltk.dsw file is in *visualc* folder, not in the *vc2005 or vnet* as it was described in Stroustrup PPP dodatek D.**

**But you may use fltk.sln from vc2005 file as well (not tested).**

1. **After Creation project in Visual stidio add some .cpp file (e.g. main.cpp) otherwise you won’t see „C/C++” section in Properties.**



1. **Crucial part!!!**

**In the step Properties->Linker-> Additional dependencies when you are adding files:**

fltkd.lib wsock32.lib comctl32.lib fltkjpegd.lib fltkimagesd.lib

**make sure that every of that is separated with semicolon ‘;’. Do not erase .libs that were there**

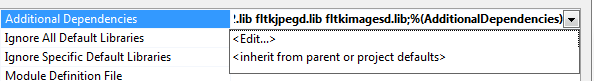
**You should have something like this:**

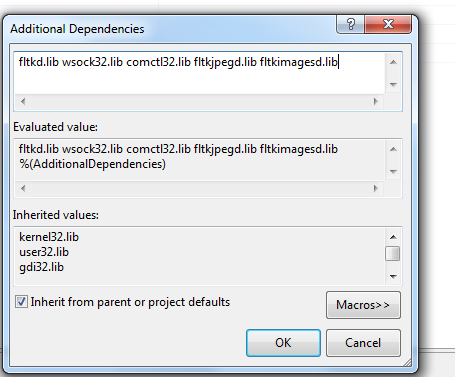
**kernel32.lib;user32.lib;gdi32.lib;winspool.lib;comdlg32.lib;advapi32.lib;shell32.lib;ole32.lib;oleaut32.lib;uuid.lib;odbc32.lib;odbccp32.lib;fltkd.lib; wsock32.lib; comctl32.lib; fltkjpegd.lib; fltkimagesd.lib;%(AdditionalDependencies)**

**Properly done section**



**If you are adding those files by Properties->Linker-> Additional dependencies->Edit**

****



**Above is wrong!**

**You should separate all .lib files with semicolon as well or with a new line.**

|  |  |
| --- | --- |
| **GOOD** | **GOOD** |

**For my comfort:**

**Properties->Linker-> Ignore specific default libraries -> add** libcd.lib

**OTHERS INSTRUCTION FROM THE INTERNET**

**This one is like in the Stroustrup PPP book**

Source: <http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html>, last access 13.04.2018

**FLTK Installation and Project setup**

**Table of Contents**

1. [Downloading FLTK](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#downloading)
2. [Installing FLTK](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#installing)
3. [Using FLTK in linux.cs.tamu.edu](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#linux)
4. [Using FLTK in Visual Studio .NET](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#project)
5. [Test if it worked](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#test)

**Downloading FLTK**

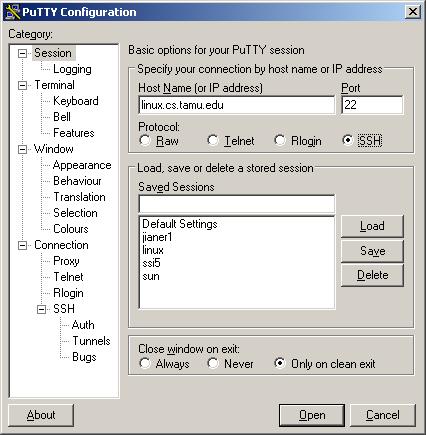
1. Go to <http://fltk.org>.
2. Click Software in the navigation menu.
3. Choose "FLTK 1.1.x" in the dropdown and click "Show Download Locations".
4. Choose a download location and download the .zip file.

**Installing FLTK**

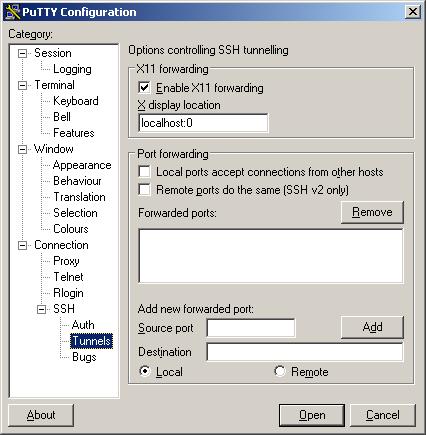
1. Unzip the downloaded file and open the main folder, fltk-1.1.?. In the visualc folder, open fltk.dsw. If asked about updating old project files, choose "Yes to All".
2. From the Build menu, choose "Build Solution". This may take a few minutes. The source code is being compiled into static link libraries so that you do not have to recompile the FLTK source code anytime you make a new project. When the process has finished, close Visual Studio.
3. From the main FLTK directory open the lib folder. Copy all the .lib files except README.lib (there should be 4) into C:\Program Files\Microsoft Visual Studio .NET\Vc7\lib
4. Go back to the FLTK main directory and copy the FL folder into C:\Program Files\Microsoft Visual Studio .NET\Vc7\include

**Using FLTK on linux.cs.tamu.edu**

1. If you are working on a windows xp PC in the lab
   * start X-win32 by click "start->All Programs->Communications->Xwin32 5.4->X-win32"
   * open putty, set hostname to "linux.cs.tamu.edu" and protocol to SSH



* + click on "Connection\SSH\Tunnels" on the left side of the putty configuration window
  + make sure "Enable X11 forwarding" is checked



* + hit "Open" to log on to linux.cs.tamu.edu using your CS userid and password

1. Put your program files and the files you downloaded from the course website in the same folder
2. Compile and link your program, suppose we want to compile sample\_main3.cpp,

g++ sample\_main3.cpp Graph.cpp Window.cpp GUI.cpp -o sample\_main3 -lfltk -lfltk\_images

1. Now you can execute your program

./sample\_main3

**Using FLTK in Visual Studio .NET**

1. Create a new project in Visual Studio .NET with one change to the usual procedure: create a "Windows application" instead of a "Console application" when choosing your project type.
2. In the solution explorer side-window, right click the project icon (not the solution icon) and choose Properties.
3. In the Properties dialog box, in the left menu, click the "Linker" folder. This expands a submenu. In this submenu, click "Input". In the text field on the right, enter the following text:

fltkd.lib wsock32.lib comctl32.lib fltkjpegd.lib fltkimagesd.lib

In the Ignore Specific Library text field, enter the following text:

libcd.lib

1. In the left menu of the same Properties window, click "C/C++" to expand a different submenu. Click the "Code Generation" submenu item. In the right menu, change the "Runtime Library" dropdown to "Multi-threaded Debug DLL (/MDd)". Click "Ok" to close the Properties window.
2. **(For CPSC labs in HRBB only)** Go to the Tools menu and select Options. Then go to Projects and Solutions -> VC++ Directories. Change the "Show directories for:" combo box to "Include Files". Add a new directory by clicking the New Line button (looks like a folder). Add "C:\fltk-1.1.7" to the list. Now change the "Show directories for:" combo box to "Library Files" and add "C:\fltk-1.1.7\lib" to the list. Click OK to save.

**Testing if it all worked**

* Create a single new .cpp file in your newly created project and enter the following code. It should be able to compile without problems.

#include <FL/Fl.h>

#include <FL/Fl\_Box.h>

#include <FL/Fl\_Window.h>

int main() {

Fl\_Window window(200, 200, "Window title");

Fl\_Box box(0,0,200,200,"Hey, I mean, Hello, World!");

window.show();

return Fl::run();

}

* If it did not work:
  + Compiler error stating a .lib file could not be found: Your problem is most likely in the [installation](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html" \l "installing) section. Pay attention to step 3, which involves putting the link libraries (.lib) files where your compiler can easily find them.
  + Compiler error stating a .h file could not be opened: Your problem is most likely in the [installation](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#installing) section. Pay attention to step 4, which involves putting the header (.h) files where your compiler can easily find them.
  + Linker error involving unresolved external symbols: Your problem is most likely in the [project](http://courses.cs.tamu.edu/daugher/cpsc121/10fall/Documentation/fltk06c.html#project) section.

**The second is working with Visual Stidio 2015 and newer FLTK library version fltk-1.3.2 (not tested I used fltk-1.1.10)**

Source: http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html, last access 13.04.2018

CIS-255 Home: <http://www.c-jump.com/bcc/c255c/c255syllabus.htm>

### Installing FLTK with Visual Studio

1. [FLTK Install with Visual Studio](http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html#W01_0010_fltk_install_with_vis)
2. [Downloading FLTK](http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html#W01_0020_downloading_fltk)
3. [Building FLTK library](http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html#W01_0030_building_fltk_library)
4. [Adding FLTK library to Visual Studio](http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html#W01_0040_adding_fltk_library_t)
5. [Testing how it all works](http://www.c-jump.com/bcc/common/Talk2/Cxx/FltkInstallVC/FltkInstallVC.html#W01_0050_testing_how_it_all_wo)

### FLTK Install with Visual Studio

* These instructions explain how to download and install Fast Light Toolkit (FLTK) for bulding Graphical User Interface applications with MSVC++.

### Downloading FLTK

* Download the source of the latest stable release version (e.g. fltk-1.3.2-source.tar.gz) from [FLTK library](http://www.google.com/?q=FLTK%20library) website. FLTK is cross-platform tool that works on Linux, MAC OS, and Windows with MSVC++ 2005, 2008, 2010, and above.
* Unzip the downloaded file. The process takes two steps: first, it unzips into a .tar file. Next, unzip the .tar file, which creates a folder named fltk-1.3.2 (or whatever version you downloaded.) For example,
* C:\Users\yourname\Downloads\fltk-1.3.2

### Building FLTK library

1. Navigate to folder
2. fltk-1.3.2\ide\VisualC2010
3. Double-click the file named fltk.sln. It opens a large MSVC++ solution that includes FLTK demo apps and the library itself. Be patient, as it takes almost a minute to load all parts into Visual Studio development environment.
4. Since FLTK is distributed as open source software, you must build the library on our own machine. Go to Visual Studio menu
5. Build -> Configuration Manager...

and switch active configuration to **Release**.

1. Click Visual Studio menu
2. Build -> Rebuild Solution

Be patient as the surce code is being compiled. It will take several minutes to finish. If everything is okay, you should see a message similar to

========== Rebuild All: 79 succeeded, 0 failed, 0 skipped ==========

at the end.

1. Exit Visual Studio.

### Adding FLTK library to Visual Studio

* Once built, FLTK can be added to Visual Studio development environment. To begin, locate Visual C++ installation folder on your local machine. In most cases it is
* On 32-bit machine: C:\Program Files\Microsoft Visual Studio 10.0\VC
* On 64-bit machine: C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC
* There are two steps:
  1. Copy FL and GL **folders**, for example,
  2. C:\Users\yourname\Downloads\fltk-1.3.2\FL
  3. C:\Users\yourname\Downloads\fltk-1.3.2\GL

to the include folder of Visual Studio:

On 32-bit machine: C:\Program Files\Microsoft Visual Studio 10.0\VC\include

On 64-bit machine: C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\include

* 1. Copy **all files** from the lib folder,
  2. C:\Users\yourname\Downloads\fltk-1.3.2\lib

to the lib folder of Visual Studio:

On 32-bit machine: C:\Program Files\Microsoft Visual Studio 10.0\VC\lib

On 64-bit machine: C:\Program Files (x86)\Microsoft Visual Studio 10.0\VC\lib



### Testing how it all works

1. Start Visual Studio again.
2. Click
3. File -> New -> Project

Select **Win32 Project** for the project type. Choose a name, e.g. fltk\_win32\_app, and change project location to

C:\CIS255\Projects

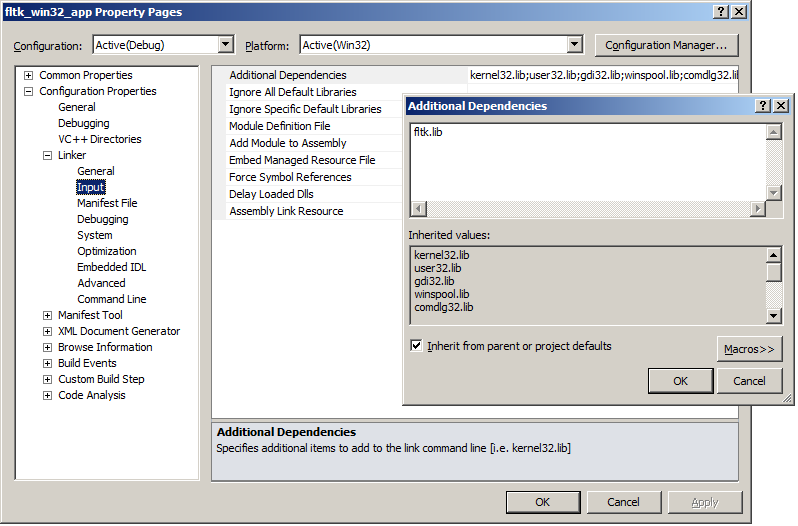
or something similar. Click **Next** on the project wizard screen and check "Empty project" option, then **Finish**.

1. Click
2. View -> Solution Explorer
3. Right-click on **Source Files** and choose
4. Add -> New item...

Use **C++ File (.cpp)** for the type of the file. Name it main.cpp.

1. The empty main.cpp file automatically opens in the editor. Copy and paste the following code:
2. **#include** <Windows.h> // include Windows.h only if using WinMain
3. **#include** <FL/Fl.H>
4. **#include** <FL/Fl\_Box.H>
5. **#include** <FL/Fl\_Window.H>
6. **// Use standard main to have console background:**
7. **// int main()**
8. **// Use WinMain if you don't want the console in the background:**
9. **int** \_\_stdcall WinMain(
10. \_\_in HINSTANCE hInstance,
11. \_\_in\_opt HINSTANCE hPrevInstance,
12. \_\_in LPSTR lpCmdLine,
13. \_\_in **int** nShowCmd
14. )
15. {
16. Fl\_Window window( 200, 200, "My window title" );
17. Fl\_Box box( 0, 50, 200, 20, "Hello" );
18. window.show();
19. **return** Fl::run();
20. }
21. In the Solution Explorer, right-click **fltk\_win32\_app** project and select **Properties**. You need to make the following three changes:
22. Configuration Properties
23. C/C++
24. Code Generation
25. Runtime Library
26. <<Edit..>>
27. and set it to Multi-threaded DLL (/MD)
28. Linker
29. Input
30. Additional Dependecies
31. <<Edit..>>
32. and enter fltk.lib
33. Ignore Specific Default Libraries
34. <<Edit..>>
35. and enter libcd.lib

For example,



Click **OK** and save your changes.

1. Click Visual Studio menu
2. Build -> Rebuild Solution
3. Press F5 to run the program.
4. Everything should compile and run without any problems. If you do get an error, send me an email, so I can look at the issue.