

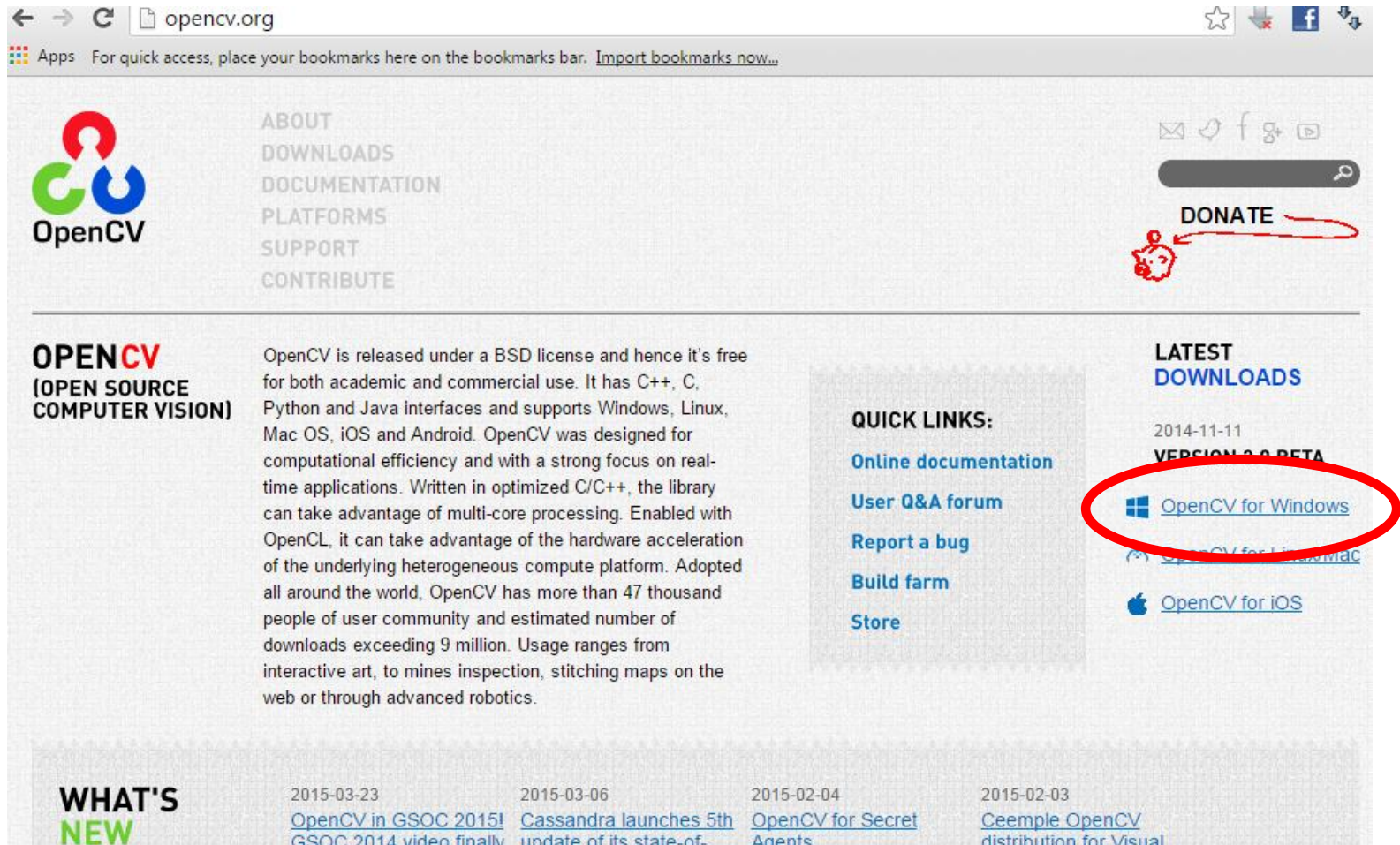
Lab Task 6.1 – const correctness

- Modify the interface `rectangle.h` such that it is `const` correct
- What to upload
 - Const corrected file with one inline comment for each usage of `const` keyword

Working with libraries

- Large programs come as a collection of libraries
- Each library has
 - Application programmer interface (API)
 - Header files
 - Pre-compiled libraries
 - static -> <name>.lib
 - dynamic -> <name>.dll
 - Source code (for Open Source projects)

Example – OpenCV



The screenshot shows the OpenCV.org website. The browser address bar displays 'opencv.org'. The website header includes the OpenCV logo, a navigation menu with links like 'ABOUT', 'DOWNLOADS', 'DOCUMENTATION', 'PLATFORMS', 'SUPPORT', and 'CONTRIBUTE', and a 'DONATE' button with a red piggy bank icon. The main content area features a large text block about OpenCV's BSD license and multi-platform support. To the right, a 'QUICK LINKS' section lists 'Online documentation', 'User Q&A forum', 'Report a bug', 'Build farm', and 'Store'. Below this, the 'LATEST DOWNLOADS' section shows the date '2014-11-11' and 'VERSION 2.2 BETA'. A red circle highlights the 'OpenCV for Windows' download link, which is accompanied by a Windows logo icon. Other links for 'OpenCV for Linux/mac' and 'OpenCV for iOS' are also visible. The footer section, titled 'WHAT'S NEW', lists recent updates with dates and links, such as 'OpenCV in GSOC 2015!', 'Cassandra launches 5th update of its state-of-the-art', 'OpenCV for Secret Agents', and 'Ceemle OpenCV distribution for Visual Studio'.

OpenCV

ABOUT
DOWNLOADS
DOCUMENTATION
PLATFORMS
SUPPORT
CONTRIBUTE

OPENCV
(OPEN SOURCE
COMPUTER VISION)




OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform. Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 9 million. Usage ranges from interactive art, to mines inspection, stitching maps on the web or through advanced robotics.

QUICK LINKS:

- [Online documentation](#)
- [User Q&A forum](#)
- [Report a bug](#)
- [Build farm](#)
- [Store](#)

LATEST DOWNLOADS

2014-11-11
VERSION 2.2 BETA

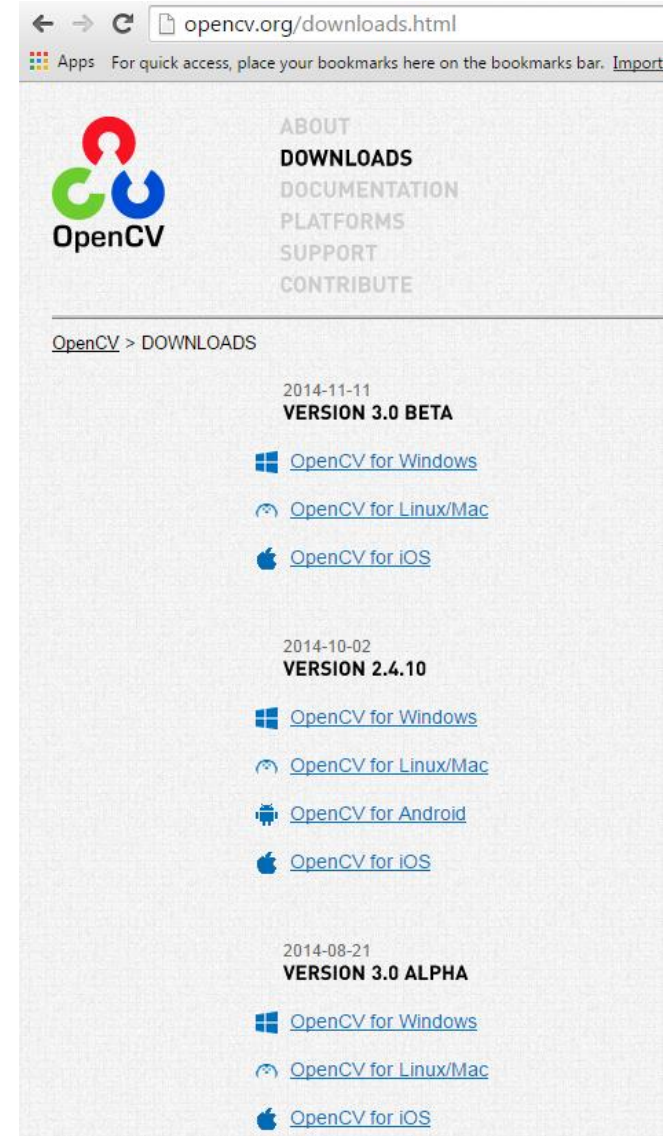
-  [OpenCV for Windows](#)
-  [OpenCV for Linux/mac](#)
-  [OpenCV for iOS](#)

WHAT'S NEW

Date	Event
2015-03-23	OpenCV in GSOC 2015!
2015-03-06	Cassandra launches 5th update of its state-of-the-art
2015-02-04	OpenCV for Secret Agents
2015-02-03	Ceemle OpenCV distribution for Visual

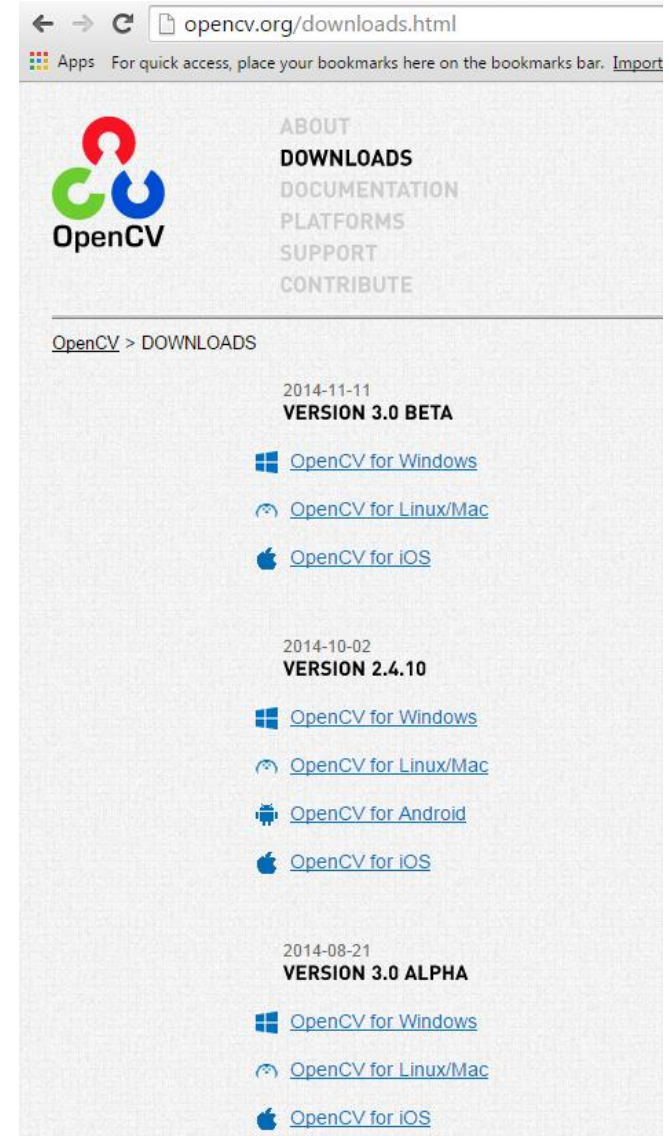
Software Releases

- Each source release is numbered with the convention **major.minor.version <suffix>**
 - Major implies API changes
 - (your previous code will not work)
 - Minor implies additional features
 - Version implies small changes / bug fixes



Software Releases

- Each source release is numbered with the convention **major.minor.version <suffix>**
 - ALPHA suffix (called **alpha release**) implies major code changes that are not comprehensively tested
 - Use at your own risk
 - BETA suffix (called **beta release**) implies substantial bug fixes and partially updated documentation
 - Still use at your own risk
 - **Stable releases have no suffix**
 - Can be used in production code (AS IS warranty)



Using OpenCV

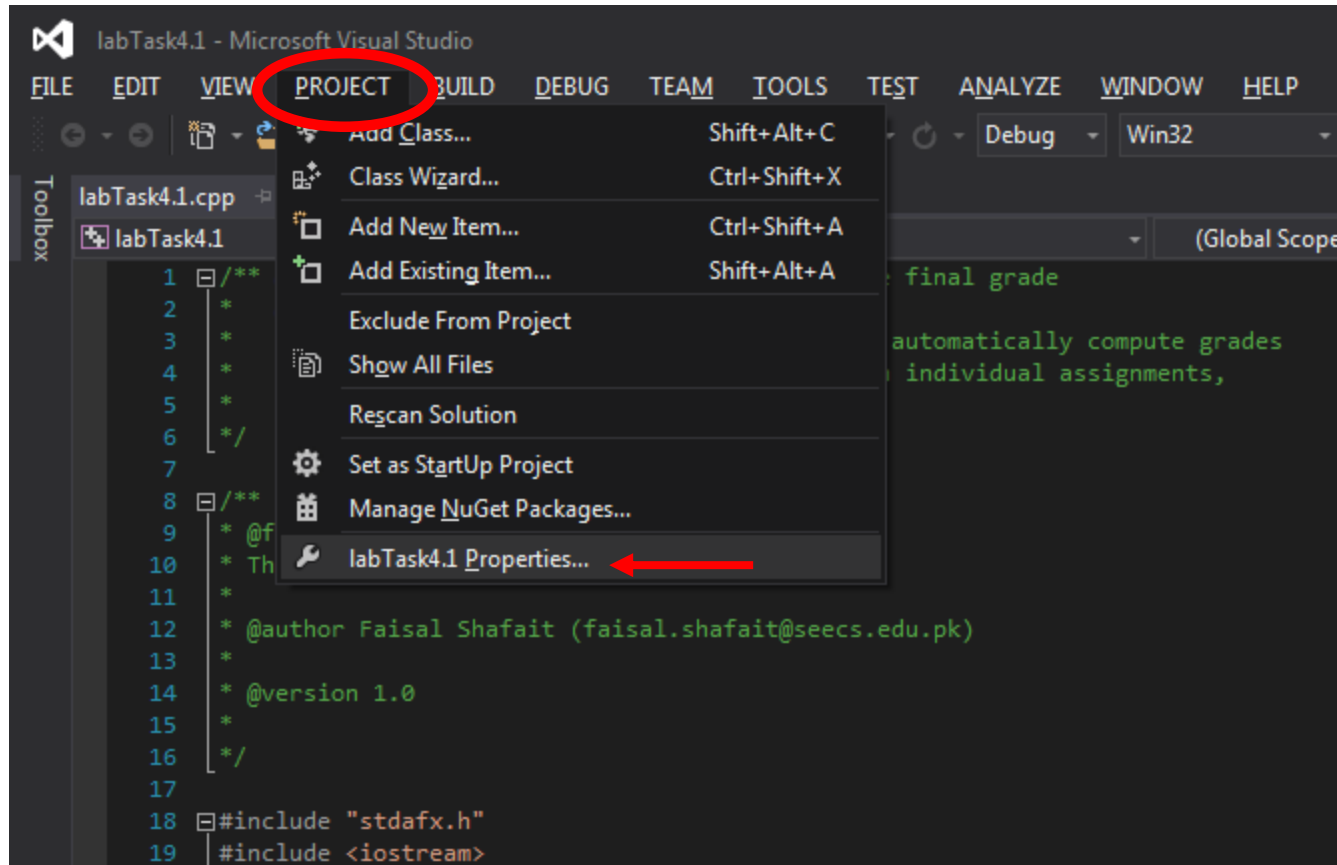
- OpenCV provides you
 - ✓ API
 - ✓ Header files
 - ✓ Pre-compiled libraries
 - ✓ static -> <name>.lib
 - ✓ dynamic -> <name>.dll
 - ✓ Source code
 - ✓ Example code (opencv\sources\samples\cpp)

Compiling code with external libraries

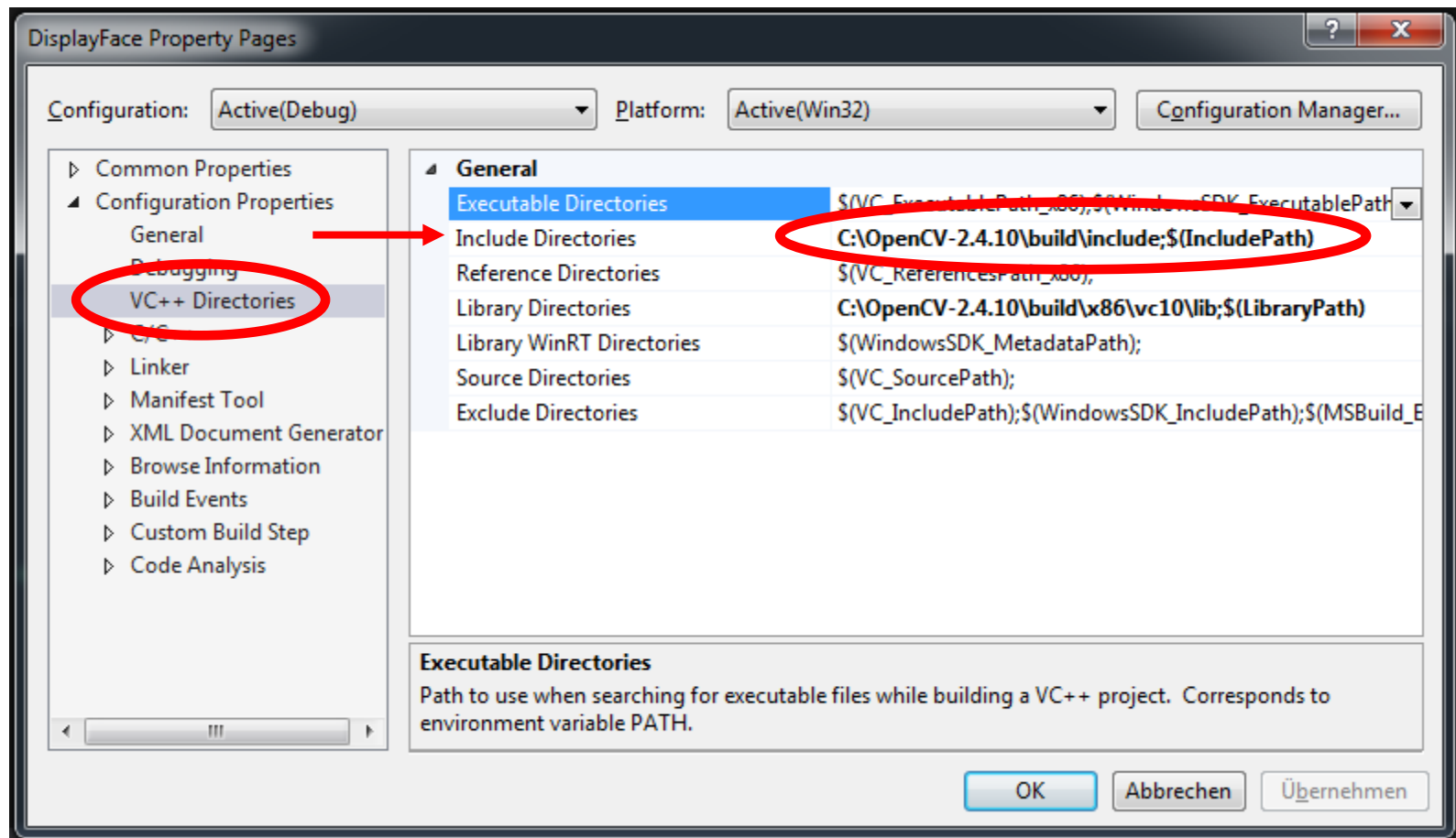
■ Four simple steps

1. Tell the compiler **where the header files are**
2. Tell the compiler **where the library files are**
3. Tell the compiler **which library files to use**
4. **Copy DLLs** to the same folder where exe is

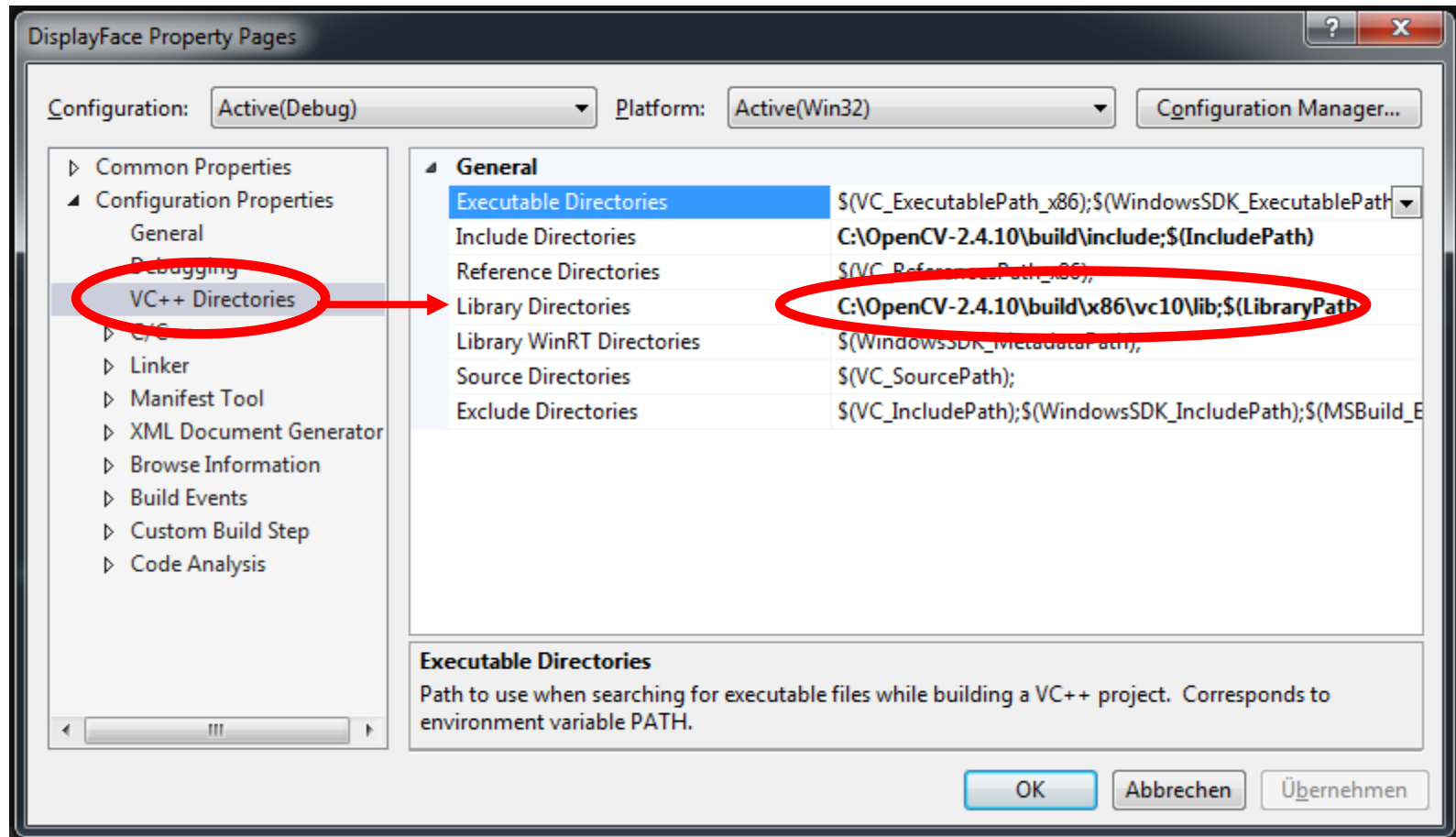
How to *tell* the compiler?



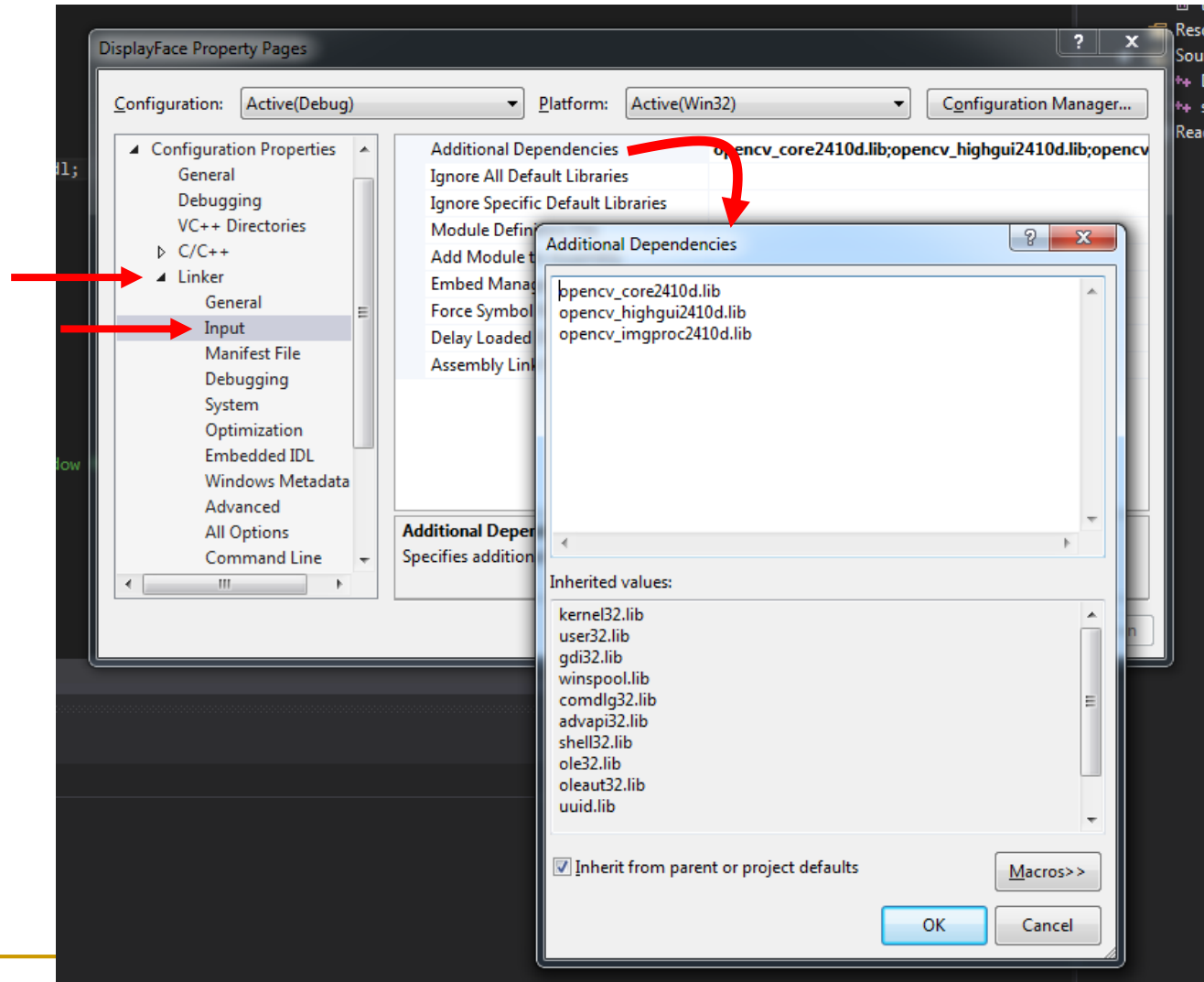
Header files path



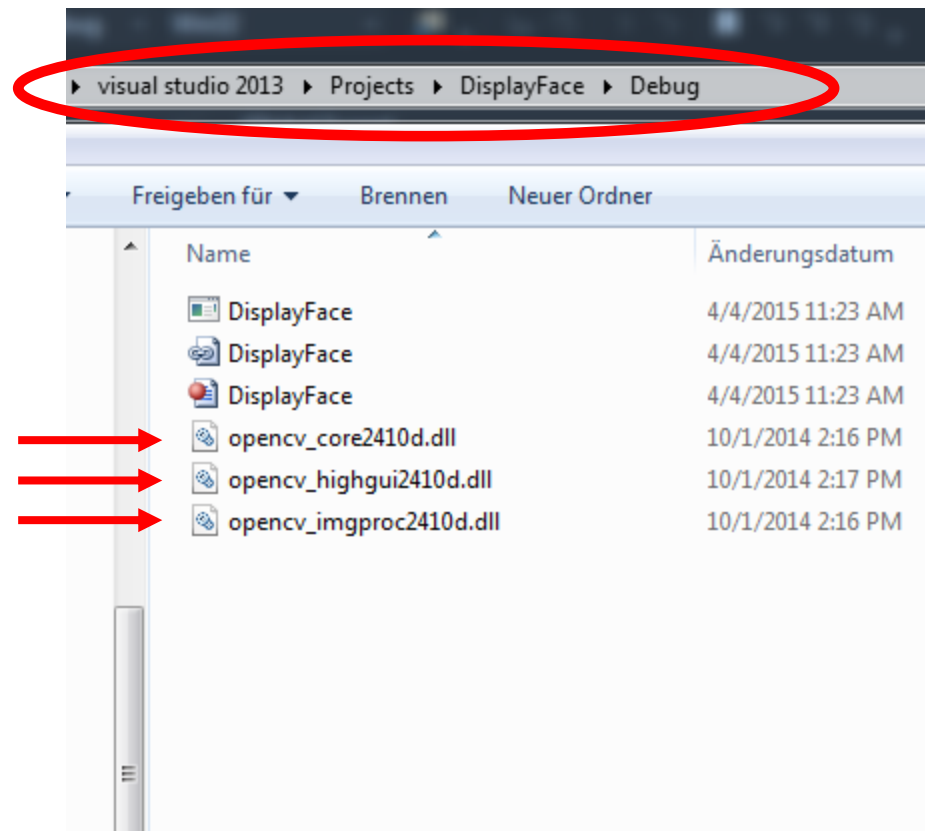
Library files path



Which libraries to use?



Copy DLLs to the folder containing the application



Lab Task 6.2 – Display your Image

- opencv\sources\samples\cpp\
 - tutorial_code\
 - introduction\
 - windows_visual_studio_Opencv
- Compile and run the code to display an image containing your face
- Upload a screen-shot of the output (sample output on next slide)

Sample Output

