C/C++ Program Design

LAB 14

CONTENTS

- Learn to use exception handling
- Installing openCV

2 Knowledge Points

2.1 Exception and Exception handling

2.2 C++ Standard Exception

2.3 Installing openCV

2.1 Exception and Exception Handling

1. What is exception?

An exception is a situation, which occurred by the runtime error. In other words, an exception is a runtime error. An exception may result in loss of data or an abnormal execution of program.

Exception handling is a mechanism that allows you to take appropriate action to avoid runtime errors.

Let’s consider a simple example:

a is divided by b, if b equals to zero, what will happen?

Example of a program without exception handling

```
#include <iostream>

using namespace std;

double Quotient(int a, int b);

int main()
{
    int a, b;
    double d;
    a = 5;
    b = 0;

    d = Quotient(a, b);
        cout << "The quotient of " << a << "/" << b << " is:" << d << endl;


    return 0;
}
double Quotient(int a, int b)
{
    return (double)a / b;
}
```

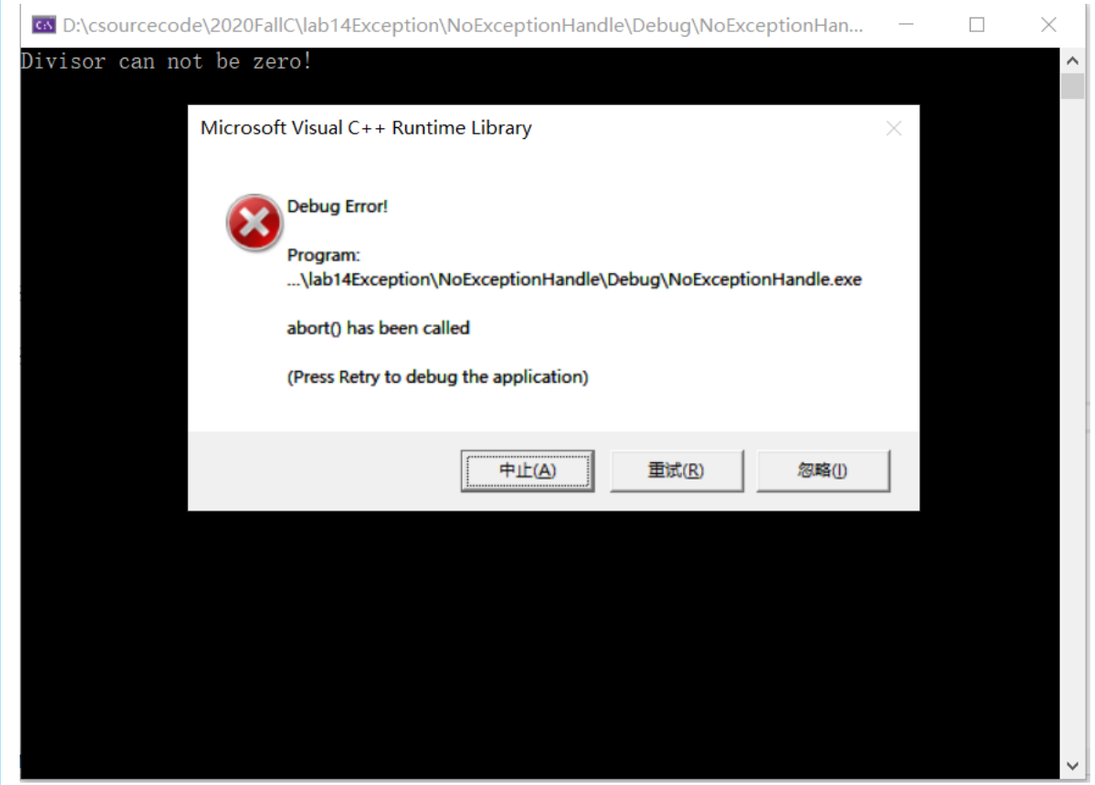
```
The qeotient of 5/0 is:inf
```

When divisor is zero, compiler generates a special floating-point value that represents infinity; ***cout*** displays this value as ***Inf, inf or INF***.

Example of a program with if statement to judge whether the divisor is zero

```
#include <iostream>

using namespace std;

double Quotient(int a, int b);

int main()
{
    int a, b;
    double d;
    a = 5;
    b = 0;

    d = Quotient(a, b);
        cout << "The quotient of " << a << "/" << b << " is:" << d << endl;


    return 0;
}
double Quotient(int a, int b)
{
    if (b == 0)
    {
        cout << "Divisor can nto be zero!";
        abort();
    }

    return (double)a / b;
}
```

You can use exit() function.



Example of a program with the return value to judge the condition

```
#include <iostream>

using namespace std;

bool Quotient(int a, int b, int &c);

int main()
{
    int a, b, c;
    double d;
    a = 5;
    b = 0;

    if (Quotient(a, b, c))
        cout << "The quotient of " << a << "/" << b << " is:" << c << endl;
    else
        cout << "The divisor can not be zero!" << endl;

    return 0;
}
bool Quotient(int a, int b, int &c)
{
    if (b == 0)
        return false;
    else
    {
        c = a / b;
        return true;
    }
}
```

To judge whether the divisor is zero by return value

```
The divisor can not be zero!
```

2. Exception handling

C++ provides **three keywords** to support exception handling

- **try**: The try block contain statements which may generate exceptions
- **throw**: When an exception occur in try block, it is thrown to the catch block using throw keyword
- **catch**: The catch block defines the action to be taken, when an exception occur.

The syntax for using **try/catch** as follows:

```
try {
   // protected code
} catch( ExceptionName e1 ) {
   // catch block
} catch( ExceptionName e2 ) {
   // catch block
} catch( ExceptionName eN ) {
   // catch block
}
catch(…)
{

}
```

Catches any type exception

You can list down multiple **catch** statements to catch different type of exceptions in case your **try** block raises more than one exception in different situations.

How does exception handling work

- When a problem is detected during the computation, an exception is raised by using **keyword throw**
- The raised exceptions are handled by **the catch block**. This exception handler is indicated by the **keyword catch**. **The catch constructor must be used immediately after the try block.**
- **try** block is responsible for testing the existence of exceptions.

The following figure explains more about this:

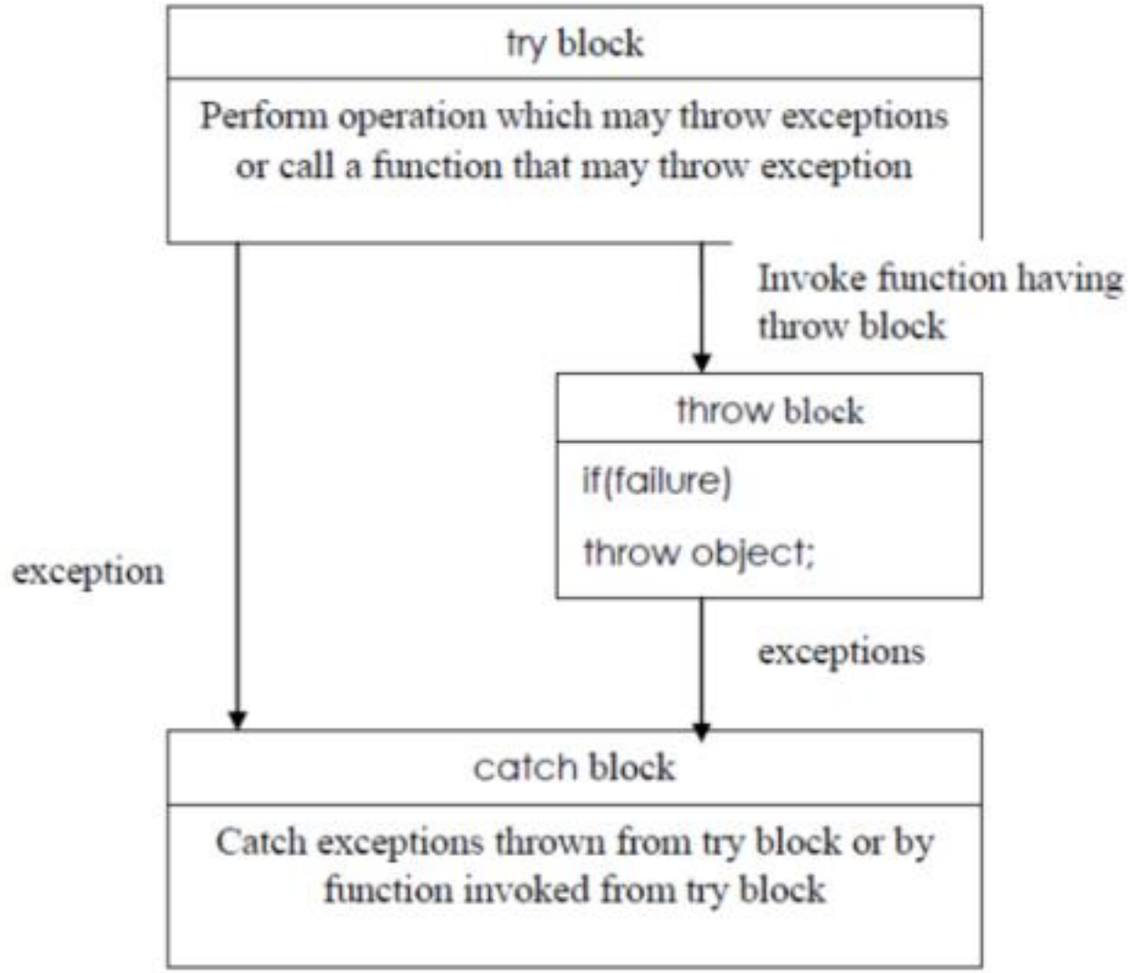


Figure: exception handling mechanism in C++

Steps taken during exception handling

1. **Hit the exception**(detect the problem causing exception)

2. **Throw the exception**(inform that an error has occurred)

3. **Catch the exception**(receive the appropriate actions)

4. **Handle the exception**(take appropriate actions)

Example of a program with exception handling using **try** and **catch**

```
#include <iostream>
using namespace std;

double Quotient(int a, int b);
int main()
{
    int a, b;
    double d;
    a = 5;
    b = 0;

    try {
        d = Quotient(a, b);
        cout << "The quotient of " << a << "/" << b << " is:" << d << endl;
    }
    catch (int x) {
        cout << "Exception : the divisor can not be zero!" << endl;
    }

    return 0;
}

double Quotient(int a, int b)
{
    if (b == 0)
        throw 0;
    return (double)a / b;
}
```

match

```
Exception: the divisor can not be zero!
```

Example of a program with exception handling using **try** and **catch**

```
#include <iostream>
using namespace std;

double Quotient(int a, int b);
int main()
{
    int a, b;
    double d;
    a = 5;
    b = 0;

    try {
        d = Quotient(a, b);
        cout << "The quotient of " << a << "/" << b << " is:" << d << endl;
    }
    catch (int code) {
        cout << "Exception : " << code << endl;
    }
    catch (const char* perror) {
        cout << perror << endl;
    }
    return 0;
}

double Quotient(int a, int b)
{
    if (b == 0)
        throw 404;
    return (double)a / b;
}
```

```
Exception : 404
```

Example of a program with exception handling using try and catch

```
#include <iostream>
using namespace std;

double Quotient(int a, int b);
int main()
{
    int a, b;
    double d;
    a = 5;
    b = 0;

    try {
        d = Quotient(a, b);
        cout << "The qeotient of " << a << "/" << b << " is:" << d << endl;
    }
    catch (int code) {
        cout << "Exception : " << code << endl;
    }
    catch (const char* perror) {
        cout << perror << endl;
    }
    return 0;
}

double Quotient(int a, int b)
{
    if (b == 0)
        throw "The divisor can not be zero!";
    return (double)a / b;
}
```

```
The divisor can not be zero!
```

Define and using exceptions

```
#include <iostream>
#include <limits>
using namespace std;

//define your exception class
class RangeError {
public:
    int iVal;
    RangeError(int _iVal) { iVal = _iVal; }
};

char to_char(int i)
{
    if (i < numeric_limits<char>::min() || numeric_limits<char>::max())
        throw RangeError(i);
    return (char)i;
}

void g(int i)
{
    try {
        char c = to_char(i);
        cout << i << " is character " << c << endl;
    }
    catch (RangeError &re) {
        cerr << "Cannot convert " << re.iVal << " to char\n" << endl;
        cerr << "Range is " << (int)numeric_limits<char>::min();
        cerr << " to " << (int)numeric_limits<char>::max() << endl;
    }
}

int main()
{
    g(-130);
    return 0;
}
```

Define your exception class

Throw the exception and invoke the constructor

Catch and handle the exception

```
Cannot convert -130 to char

Range is -128 to 127
```

Handling exceptions from a inheritance hierarchy

```
#include <iostream>
using namespace std;

class Matherr { };
class OverflowException : public Matherr { };
class UnderflowException : public Matherr { };
class ZeroDivideException : public Matherr { };

double divide(int numerator, int denominator)
{
    if(denominator == 0)
        throw ZeroDivideException();
    double d = (double) numerator / denominator;
    return d;
}
int main()
{
    try{
        cout << divide( numerator: 6,  denominator: 0) << endl;
    } catch (ZeroDivideException) {
        cerr << "Zero Divide Error" << endl;
    } catch (Matherr) {
        cerr << "Math Error" << endl;
    }

    return 0;
}
```

Output:

```
Zero Divide Error
```

Define your own exceptions by inheritance and overriding **what()** method

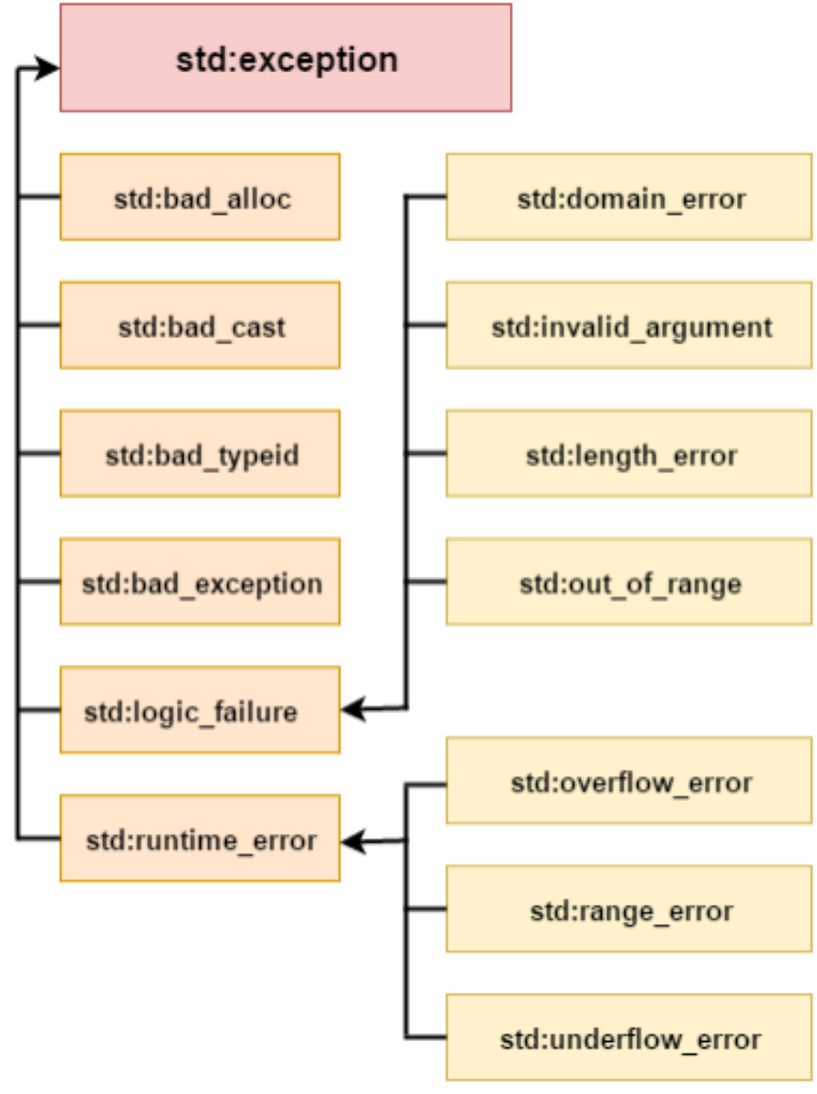
```cpp
#include <iostream>
using namespace std;

class MyException : public exception
{
public:
    const char* what()
    {
        return "C++ Exception";
    }
};

int main()
{
    try {
        throw MyException();
    }
    catch (MyException& e) {
        cout << "MyException caught" << endl;
        cout << e.what() << endl;
    }
    catch (exception &e){
        // other errors
    }

    return 0;
}
```

what() is a public method provided by exception class which returns a string and it has been overridden by all the child exception classes.

```
MyException caught
C++ Exception
```

2.2 C++ Standard Exceptions

C++ provides a list of standard exceptions defined in which we can use in our programs. These are arranged in a parent-child class hierarchy shown right.

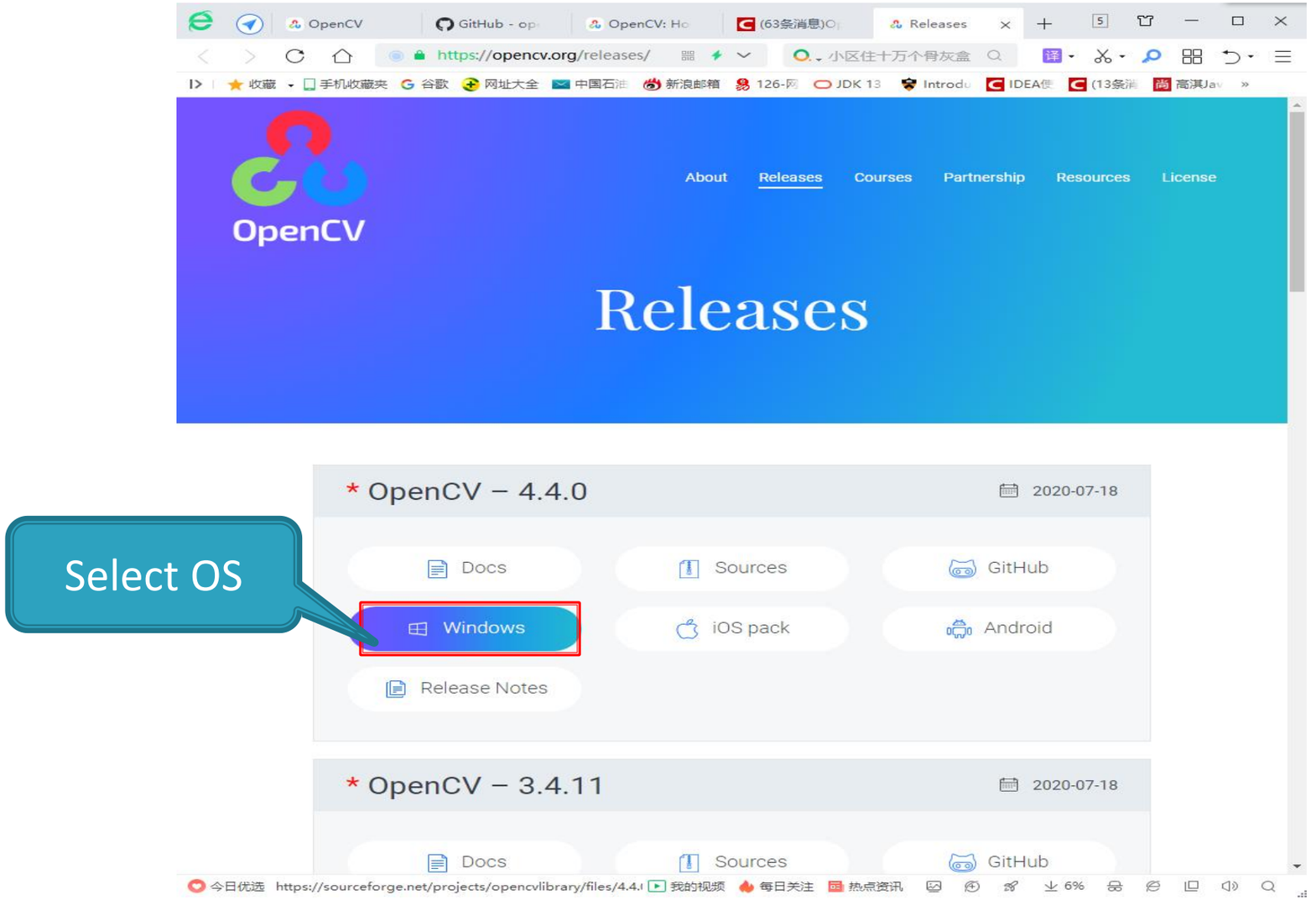


Here is the small description of each exception mentioned in the above hierarchy:

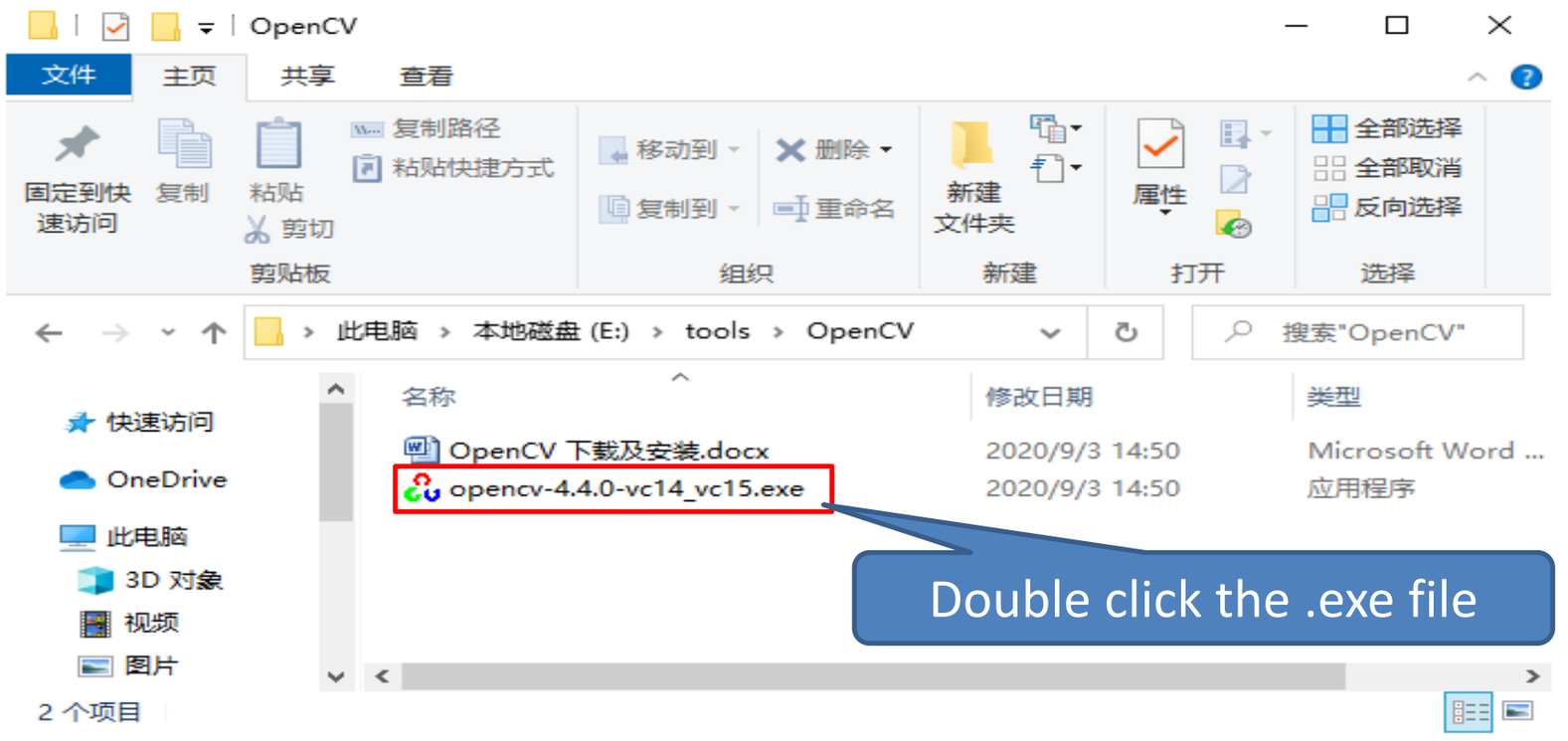
Exception	Description
std::exception	An exception and parent class of all the standard C++ exceptions.
std::bad_alloc	This can be thrown by **new**.
std::bad_cast	This can be thrown by **dynamic_cast**.
std::bad_exception	This is useful device to handle unexpected exceptions in a C++ program
std::bad_typeid	This can be thrown by typeid.
std::logic_error	An exception that theoretically can be detected by reading the code.
std::domain_error	This is an exception thrown when a mathematically invalid domain is used
std::invalid_argument	This is thrown due to invalid arguments.
std::length_error	This is thrown when a too big std::string is created
std::out_of_range	This can be thrown by the at method from for example a std::vector and std::bitset<>::operator.
std::runtime_error	An exception that theoretically can not be detected by reading the code.
std::overflow_error	This is thrown if a mathematical overflow occurs.
std::range_error	This is occured when you try to store a value which is out of range.
std::underflow_error	This is thrown if a mathematical underflow occurs.

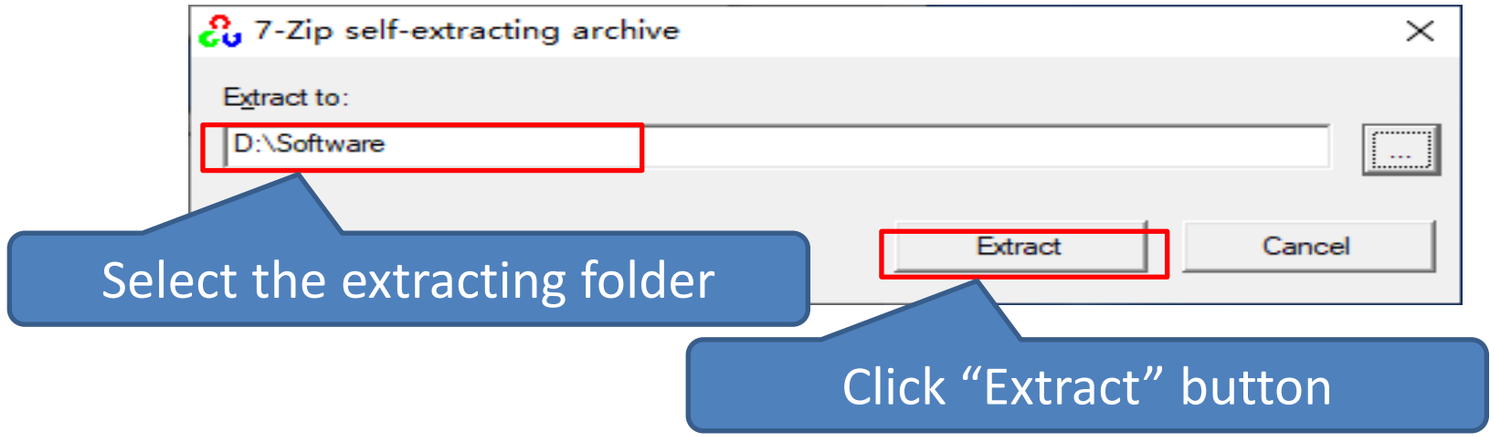
2.3 Downloading and Installing OpenCV

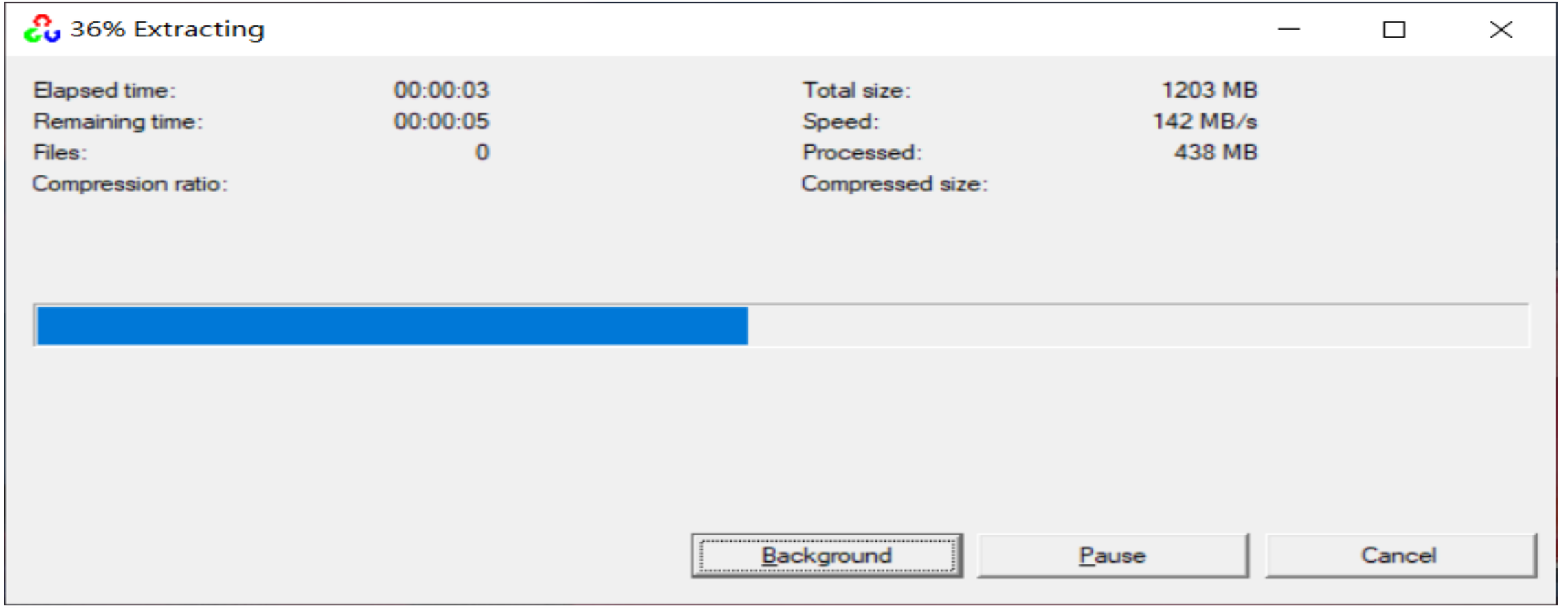
URL: https://opencv.org/releases/

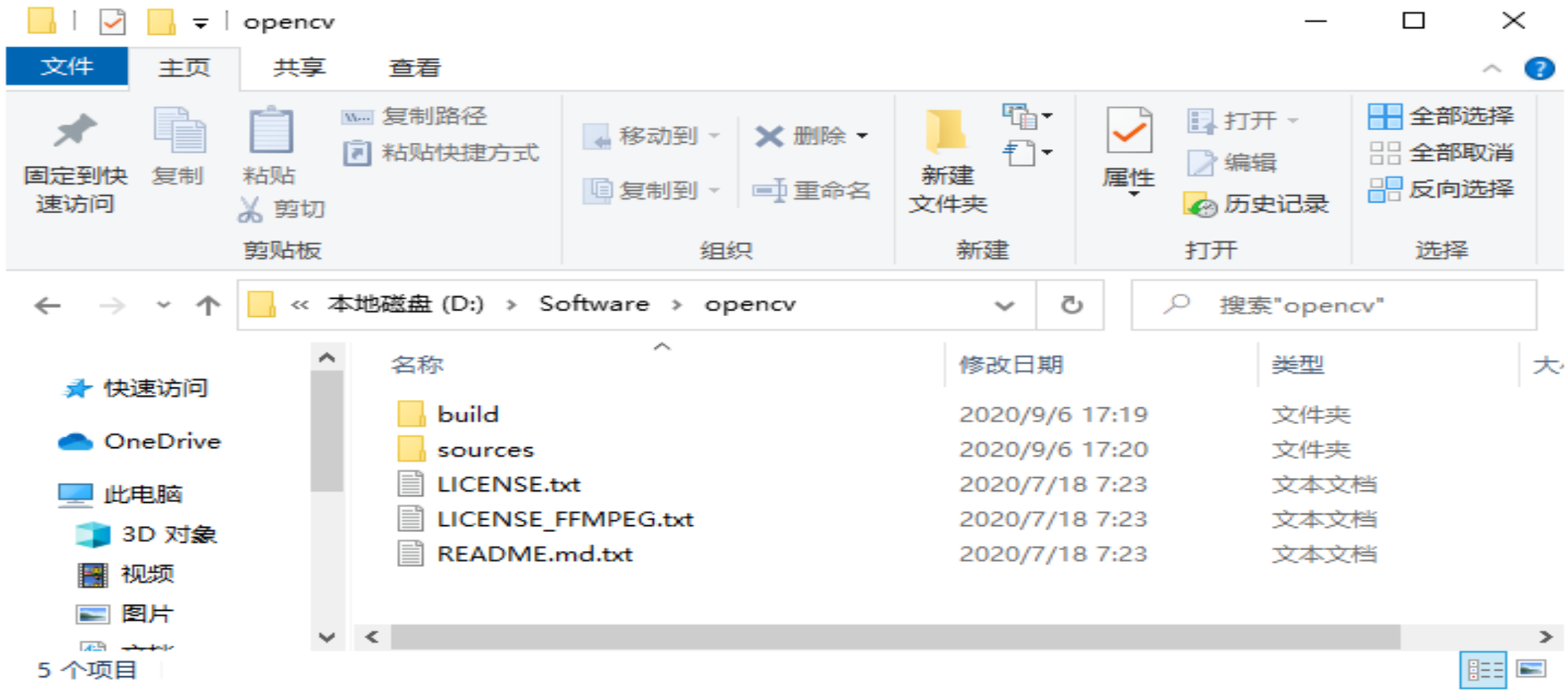


1. Extract the file

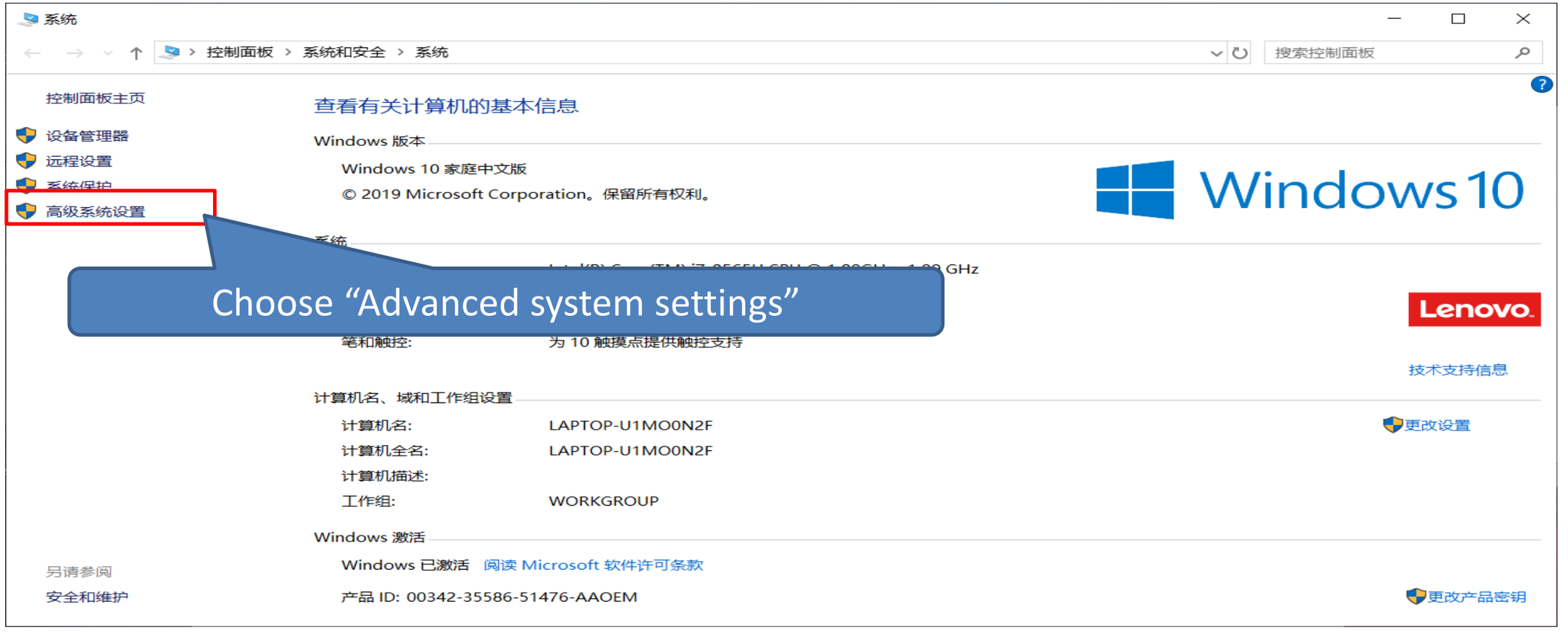



7-Zip self-extracting archive
Extract to:
D:\Software
...
Extract
Cancel
Select the extracting folder
Click “Extract” button


36% Extracting
Elapsed time: 00:00:03
Remaining time: 00:00:05
Files: 0
Compression ratio:
Total size: 1203 MB
Speed: 142 MB/s
Processed: 438 MB
Compressed size:
Background
Pause
Cancel


opencv
文件
主页
共享
查看
固定到快速访问
复制
粘贴
剪切
复制路径
粘贴快捷方式
剪贴板
移动到
复制到
删除
重命名
组织
新建文件夹
新建
属性
打开
编辑
历史记录
打开
全部选择
全部取消
反向选择
选择
本地磁盘 (D:) › Software › opencv
搜索"opencv"
快速访问
OneDrive
此电脑
3D 对象
视频
图片
名称
修改日期
类型
build
2020/9/6 17:19
文件夹
sources
2020/9/6 17:20
文件夹
LICENSE.txt
2020/7/18 7:23
文本文档
LICENSE_FFMPEG.txt
2020/7/18 7:23
文本文档
README.md.txt
2020/7/18 7:23
文本文档
5 个项目

2. Configure the environment variable




系统属性
计算机名
硬件
高级
系统保护
远程
要进行大多数更改，你必须作为管理员登录。
性能
视觉效果，处理器计划，内存使用，以及虚拟内存
设置(S)...
用户配置文件
与登录帐户相关的桌面设置
设置(E)...
启动和故障恢复
系统启动、系统故障和调试信息
设置(T)...
环境变量(N)...
Click “Environment Variable”
确定
取消
应用(A)

环境变量

liaoqm 的用户变量(U)

变量	值
IntelliJ IDEA Community Ed...	C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020...
OneDrive	C:\Users\liaoqm\OneDrive
Path	C:\Users\liaoqm\AppData\Local\Microsoft\WindowsApps;;C:\Pr...
TEMP	C:\Users\liaoqm\AppData\Local\Temp
TMP	C:\Users\liaoqm\AppData\Local\Temp

新建(N)... 编辑(E)... 删除(D)

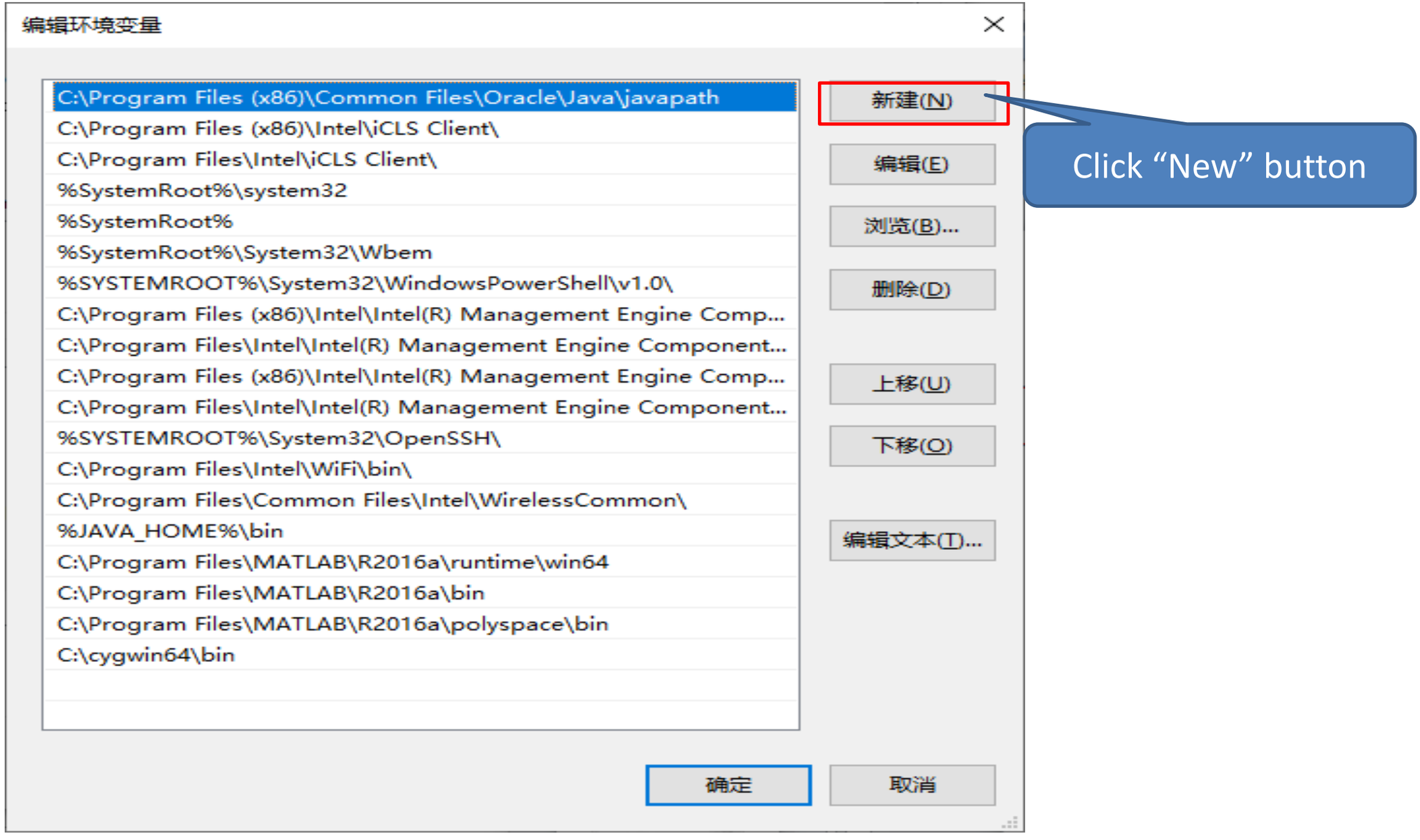
系统变量(S)

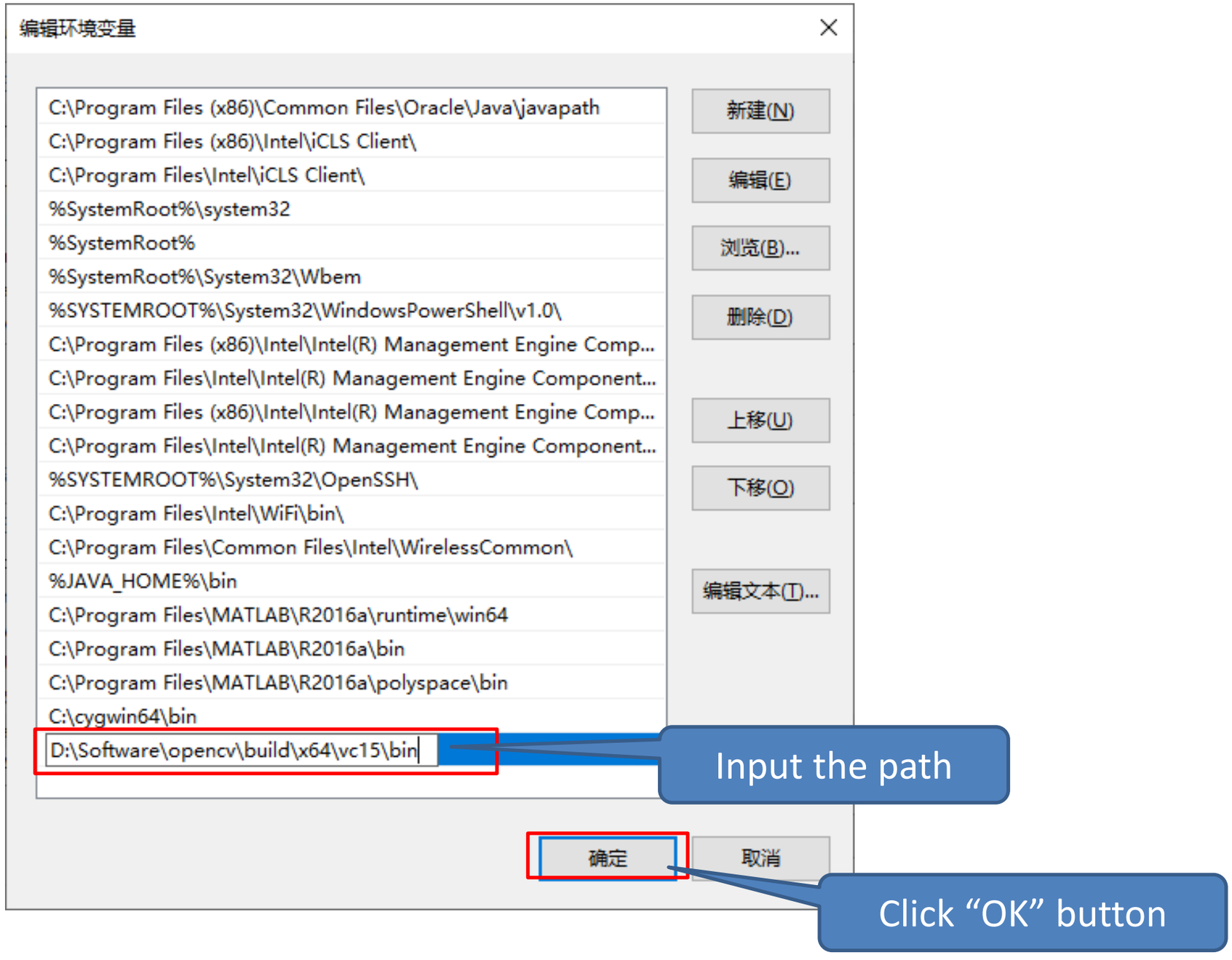
变量	值
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_221
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
Path	C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\W...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC

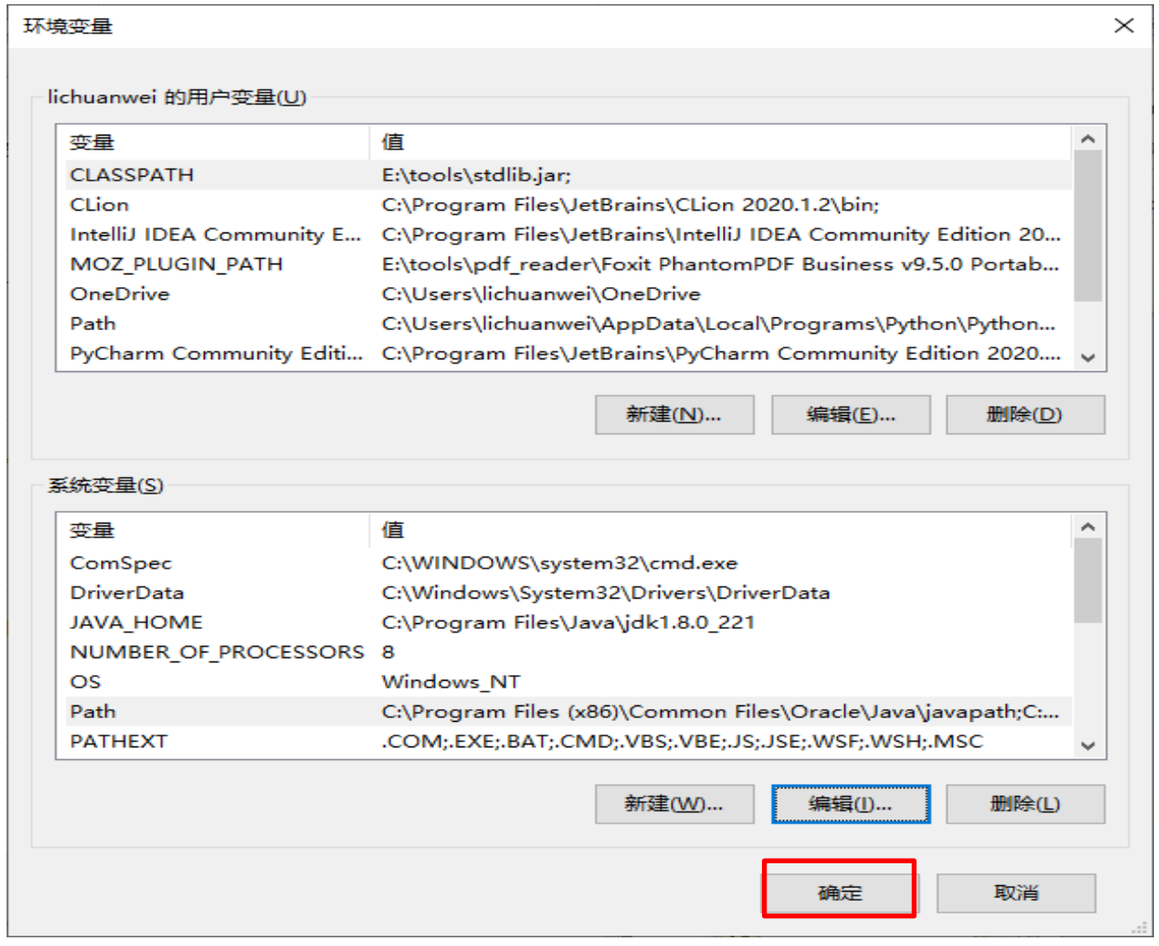
新建(W)... 编辑(I)... 删除(L)

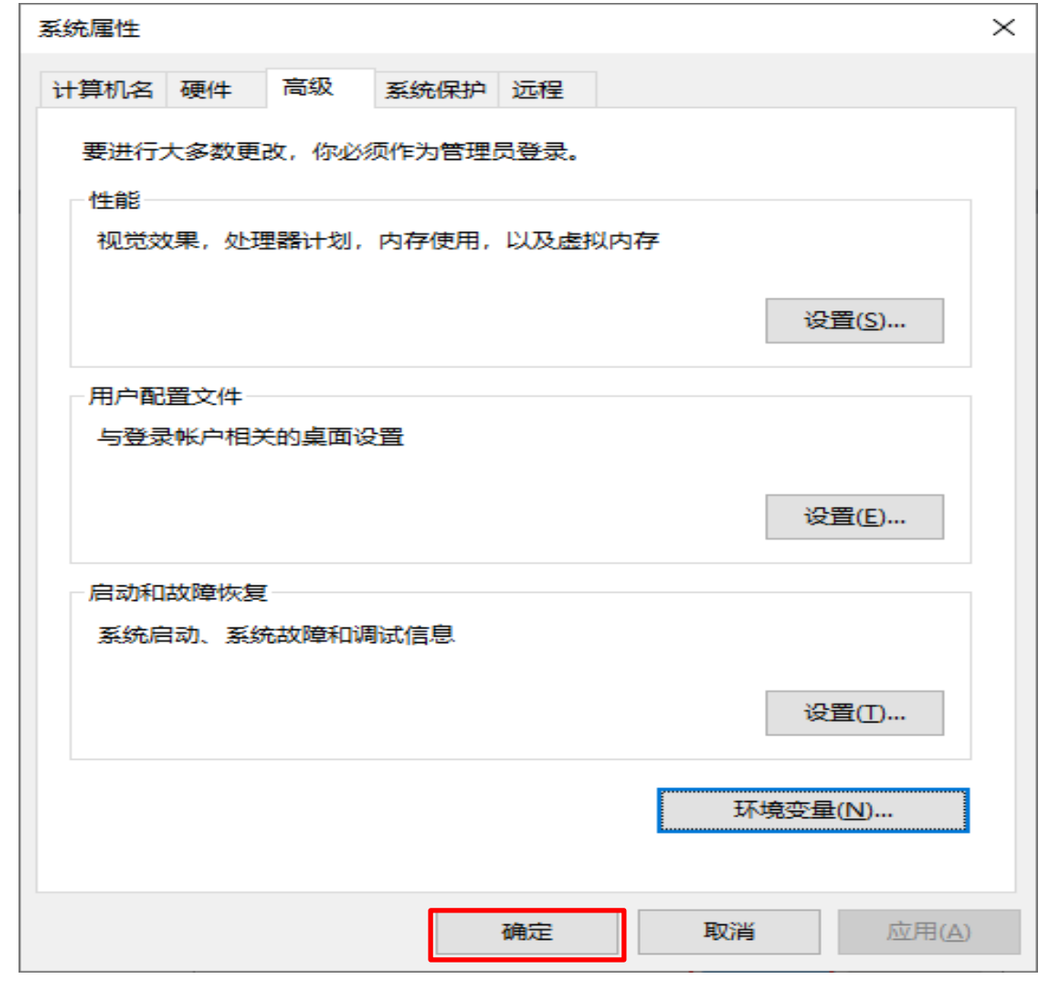
Click “Path” in system variable and click “Edit” button

确定 取消


编辑环境变量
C:\Program Files (x86)\Common Files\Oracle\Java\javapath
C:\Program Files (x86)\Intel\iCLS Client\
C:\Program Files\Intel\iCLS Client\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Comp...
C:\Program Files\Intel\Intel(R) Management Engine Component...
C:\Program Files (x86)\Intel\Intel(R) Management Engine Comp...
C:\Program Files\Intel\Intel(R) Management Engine Component...
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\Intel\WiFi\bin\
C:\Program Files\Common Files\Intel\WirelessCommon\
%JAVA_HOME%\bin
C:\Program Files\MATLAB\R2016a\runtime\win64
C:\Program Files\MATLAB\R2016a\bin
C:\Program Files\MATLAB\R2016a\polyspace\bin
C:\cygwin64\bin
新建(N)
编辑(E)
浏览(B)...
删除(D)
上移(U)
下移(O)
编辑文本(T)...
确定
取消
Click “New” button

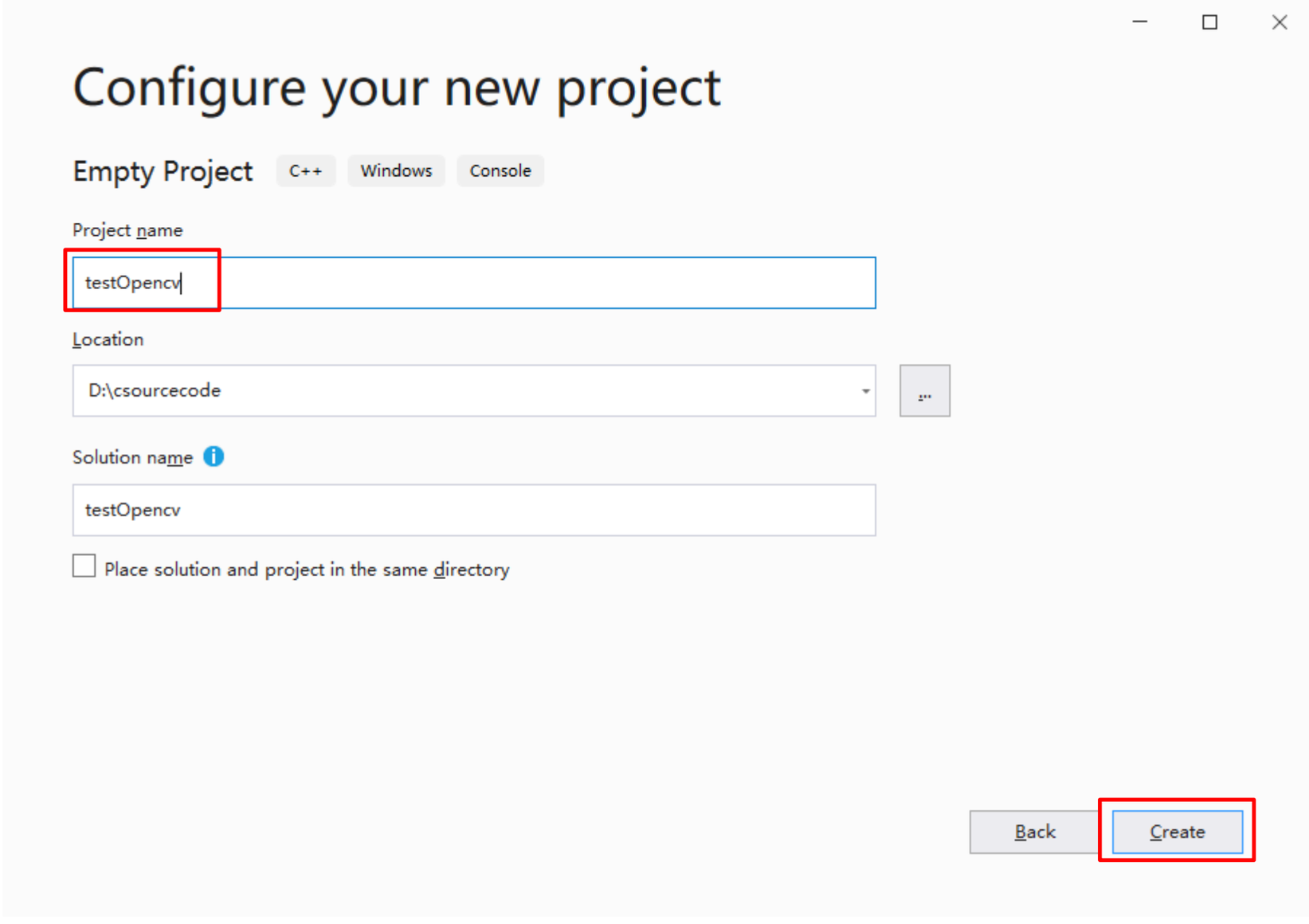

编辑环境变量
C:\Program Files (x86)\Common Files\Oracle\Java\javapath
C:\Program Files (x86)\Intel\iCLS Client\
C:\Program Files\Intel\iCLS Client\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Comp...
C:\Program Files\Intel\Intel(R) Management Engine Component...
C:\Program Files (x86)\Intel\Intel(R) Management Engine Comp...
C:\Program Files\Intel\Intel(R) Management Engine Component...
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files\Intel\WiFi\bin\
C:\Program Files\Common Files\Intel\WirelessCommon\
%JAVA_HOME%\bin
C:\Program Files\MATLAB\R2016a\runtime\win64
C:\Program Files\MATLAB\R2016a\bin
C:\Program Files\MATLAB\R2016a\polyspace\bin
C:\cygwin64\bin
D:\Software\opencv\build\x64\vc15\bin
新建(N)
编辑(E)
浏览(B)...
删除(D)
上移(U)
下移(O)
编辑文本(T)...
确定
取消
Input the path
Click “OK” button

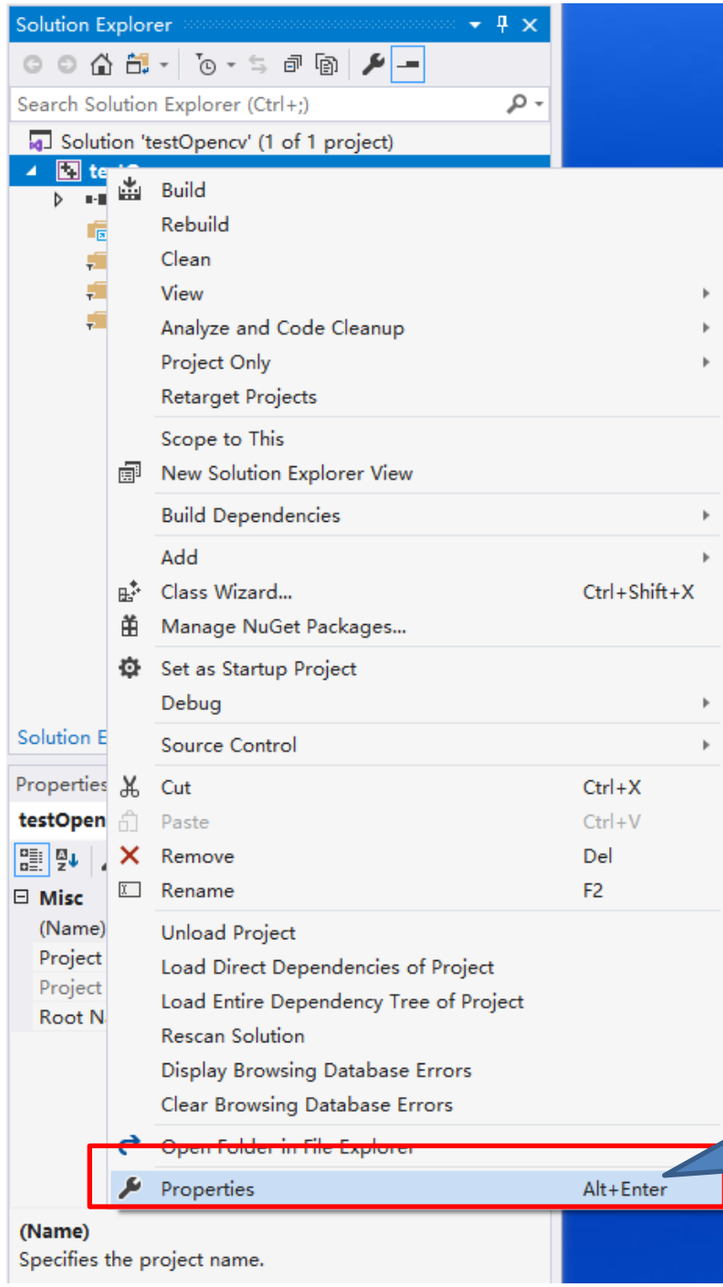

环境变量
lichuanwei 的用户变量(U)
变量 值
CLASSPATH E:\tools\stdlib.jar;
CLion C:\Program Files\JetBrains\CLion 2020.1.2\bin;
IntelliJ IDEA Community E... C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 20...
MOZ_PLUGIN_PATH E:\tools\pdf_reader\Foxit PhantomPDF Business v9.5.0 Portab...
OneDrive C:\Users\lichuanwei\OneDrive
Path C:\Users\lichuanwei\AppData\Local\Programs\Python\Python...
PyCharm Community Editi... C:\Program Files\JetBrains\PyCharm Community Edition 2020....
新建(N)...
编辑(E)...
删除(D)
系统变量(S)
变量 值
ComSpec C:\WINDOWS\system32\cmd.exe
DriverData C:\Windows\System32\Drivers\DriverData
JAVA_HOME C:\Program Files\Java\jdk1.8.0_221
NUMBER_OF_PROCESSORS 8
OS Windows_NT
Path C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:...
PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
新建(W)...
编辑(I)...
删除(L)
确定
取消


系统属性
计算机名
硬件
高级
系统保护
远程
要进行大多数更改，你必须作为管理员登录。
性能
视觉效果，处理器计划，内存使用，以及虚拟内存
设置(S)...
用户配置文件
与登录帐户相关的桌面设置
设置(E)...
启动和故障恢复
系统启动、系统故障和调试信息
设置(T)...
环境变量(N)...
确定
取消
应用(A)

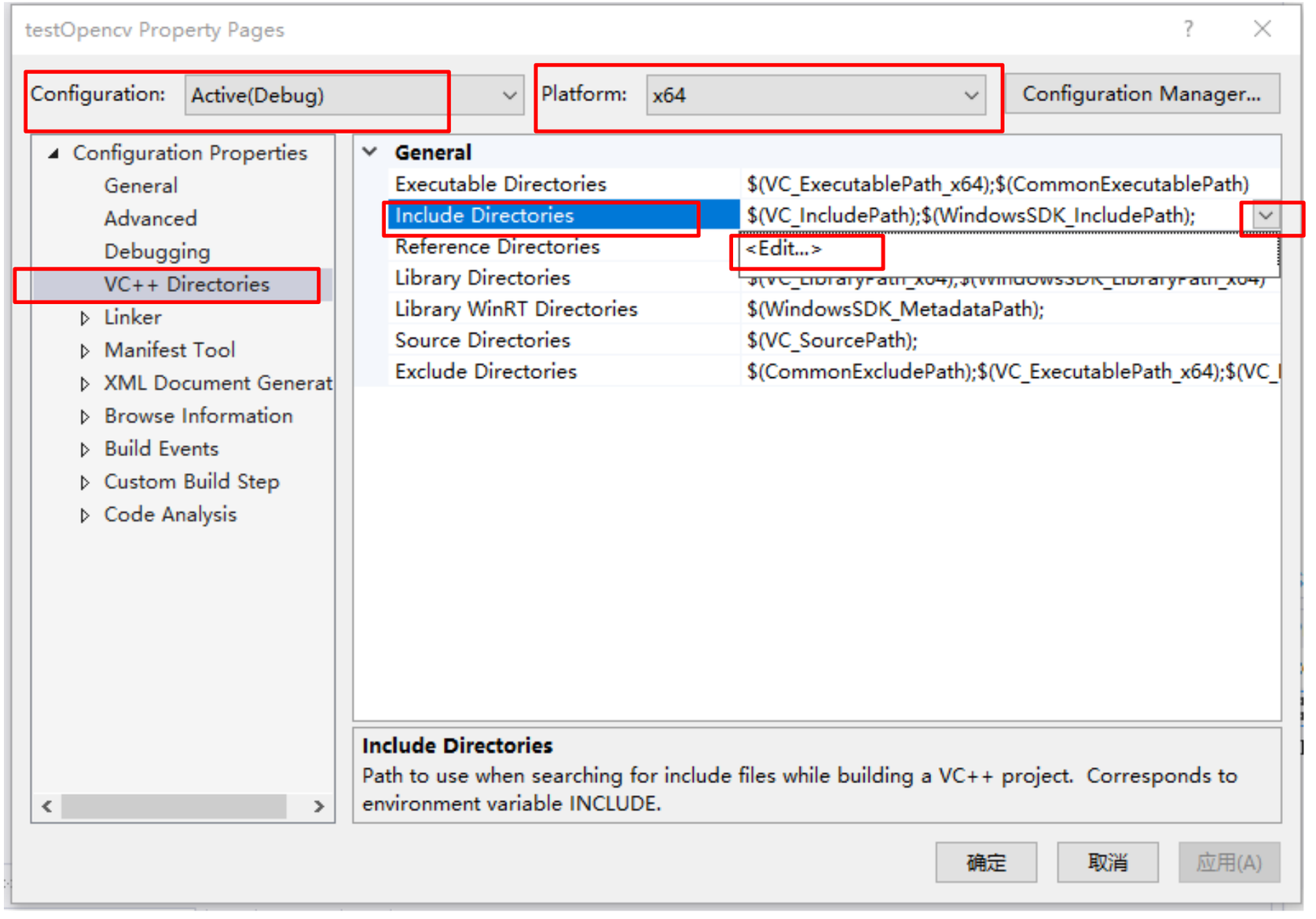
3. Configure files in VS 2019

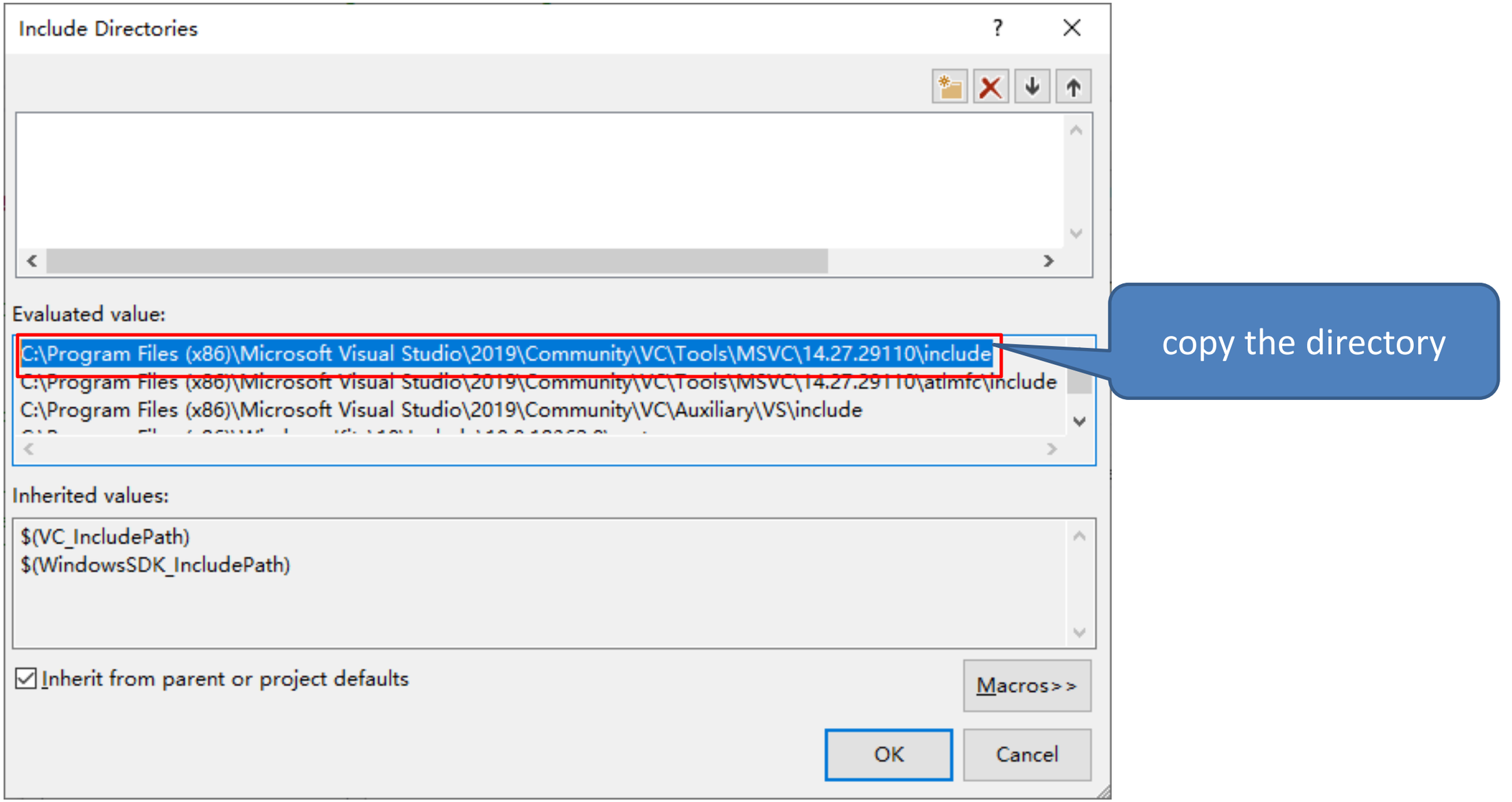
(1) Create an empty project in vs2019



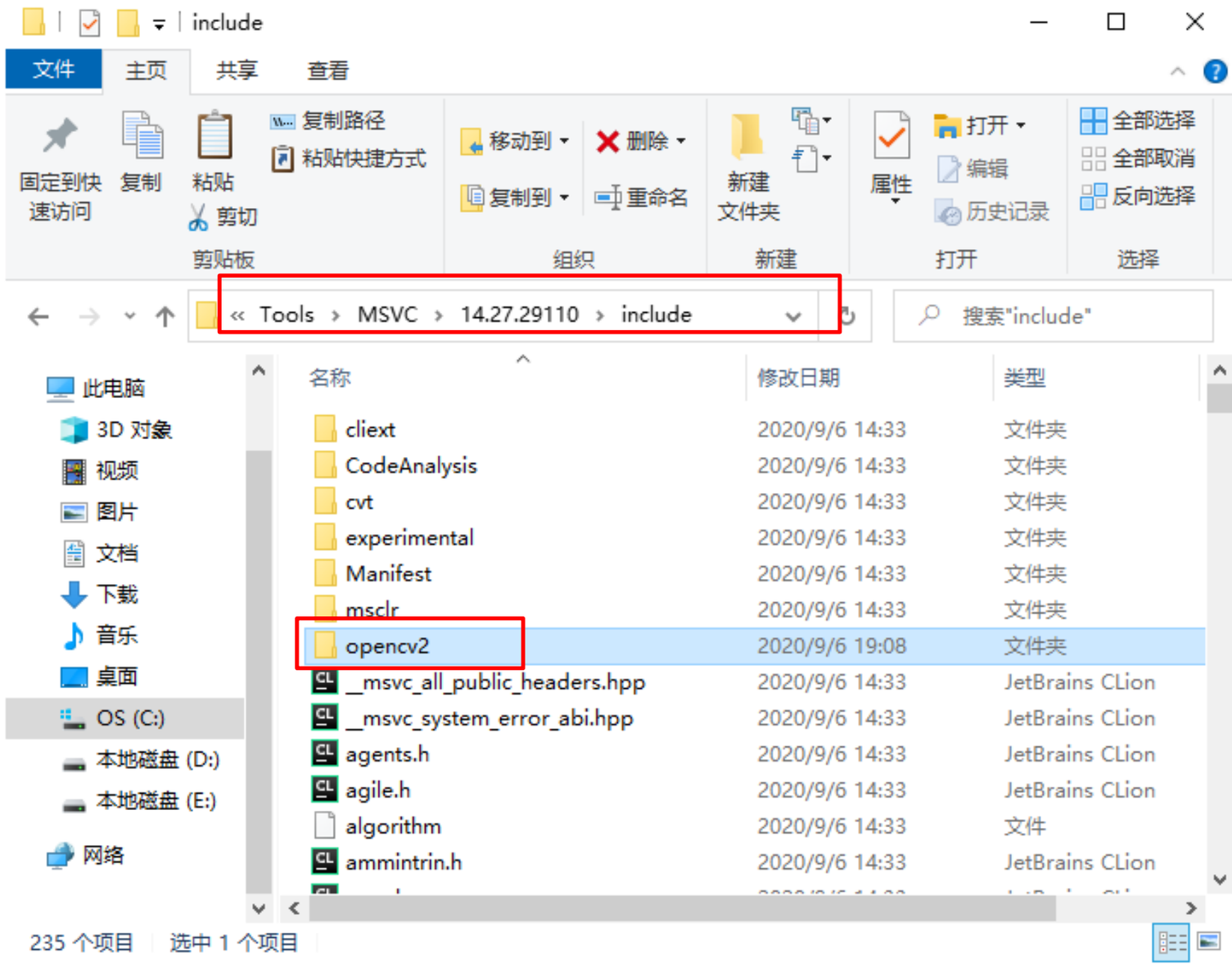


Right click the mouse at your project name and choose “Properties” menu item

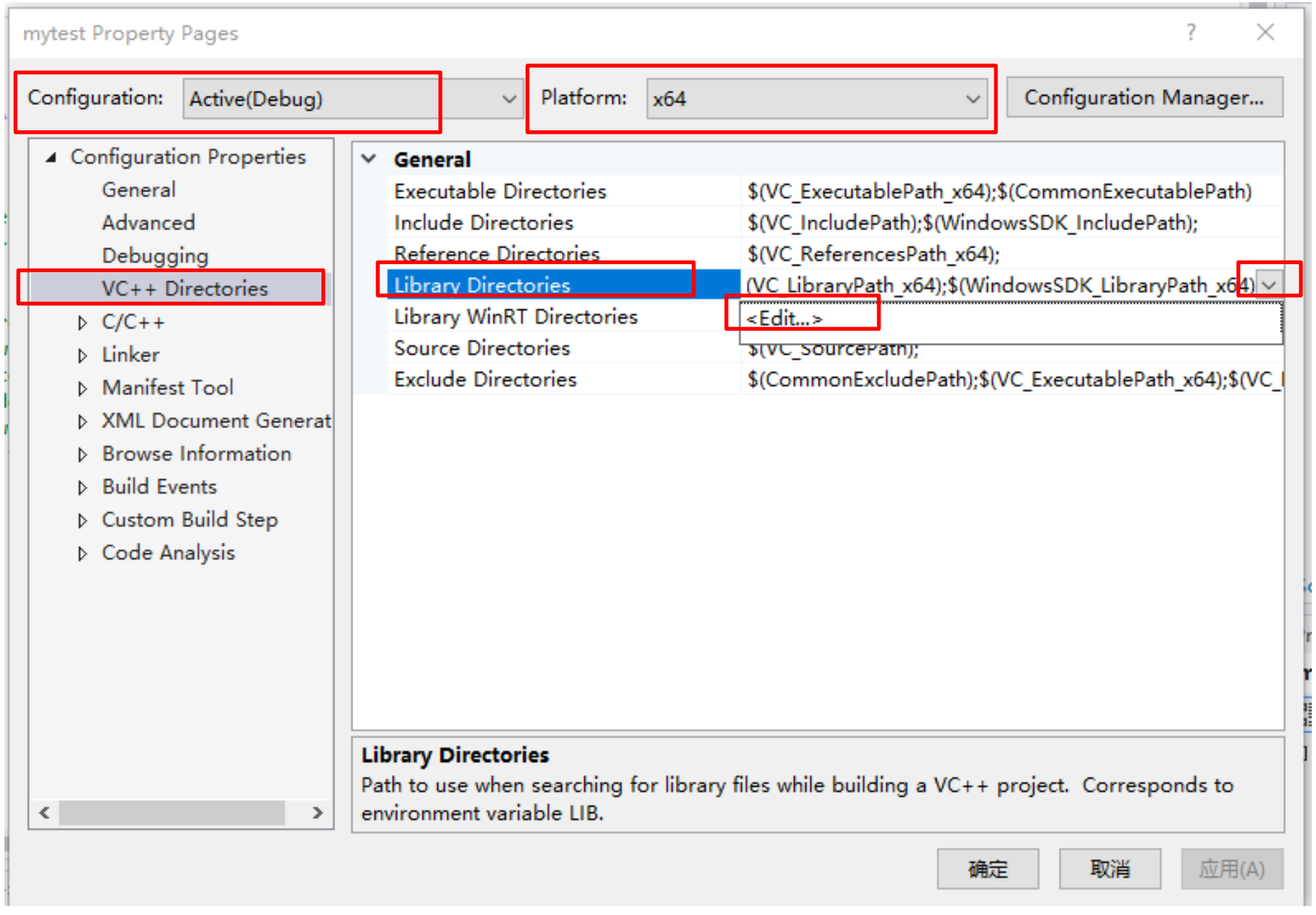

testOpencv Property Pages
Configuration: Active(Debug)
Platform: x64
Configuration Manager...
Configuration Properties
General
Advanced
Debugging
VC++ Directories
Linker
Manifest Tool
XML Document Generat
Browse Information
Build Events
Custom Build Step
Code Analysis
General
Executable Directories
$(VC_ExecutablePath_x64);$(CommonExecutablePath)
Include Directories
$(VC_IncludePath);$(WindowsSDK_IncludePath);
Reference Directories
<Edit...>
Library Directories
Library WinRT Directories
$(WindowsSDK_MetadataPath);
Source Directories
$(VC_SourcePath);
Exclude Directories
$(CommonExcludePath);$(VC_ExecutablePath_x64);$(VC_
Include Directories
Path to use when searching for include files while building a VC++ project. Corresponds to environment variable INCLUDE.
确定
取消
应用(A)

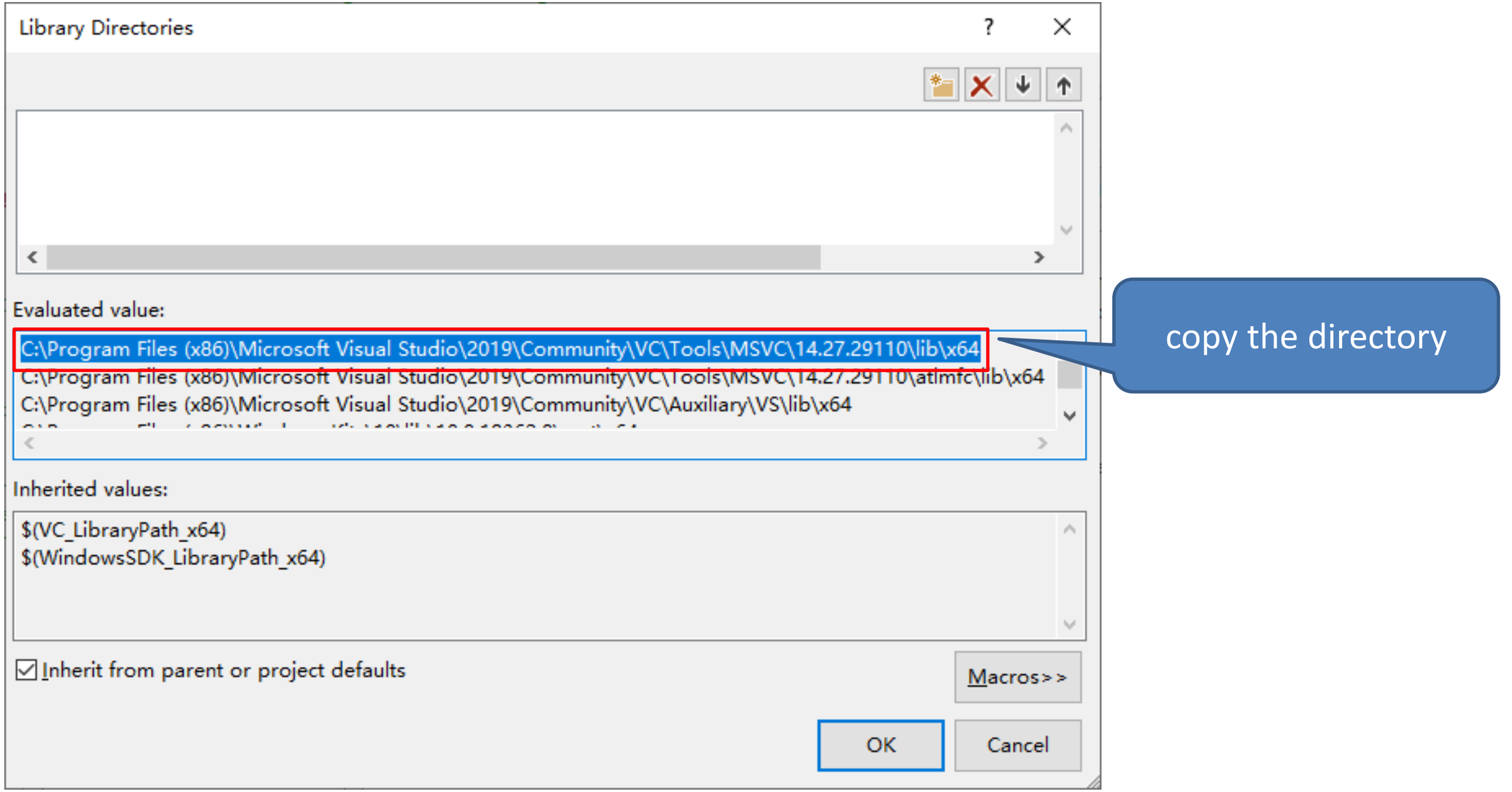

Include Directories
Evaluated value:
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.27.29110\include
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.27.29110\atlmfc\include
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\VS\include
Inherited values:
$(VC_IncludePath)
$(WindowsSDK_IncludePath)
Inherit from parent or project defaults
Macros>>
OK
Cancel
copy the directory

Copy the “opencv2” folder which is in the “D:\Software\opencv\build\include” to the directory you just copied

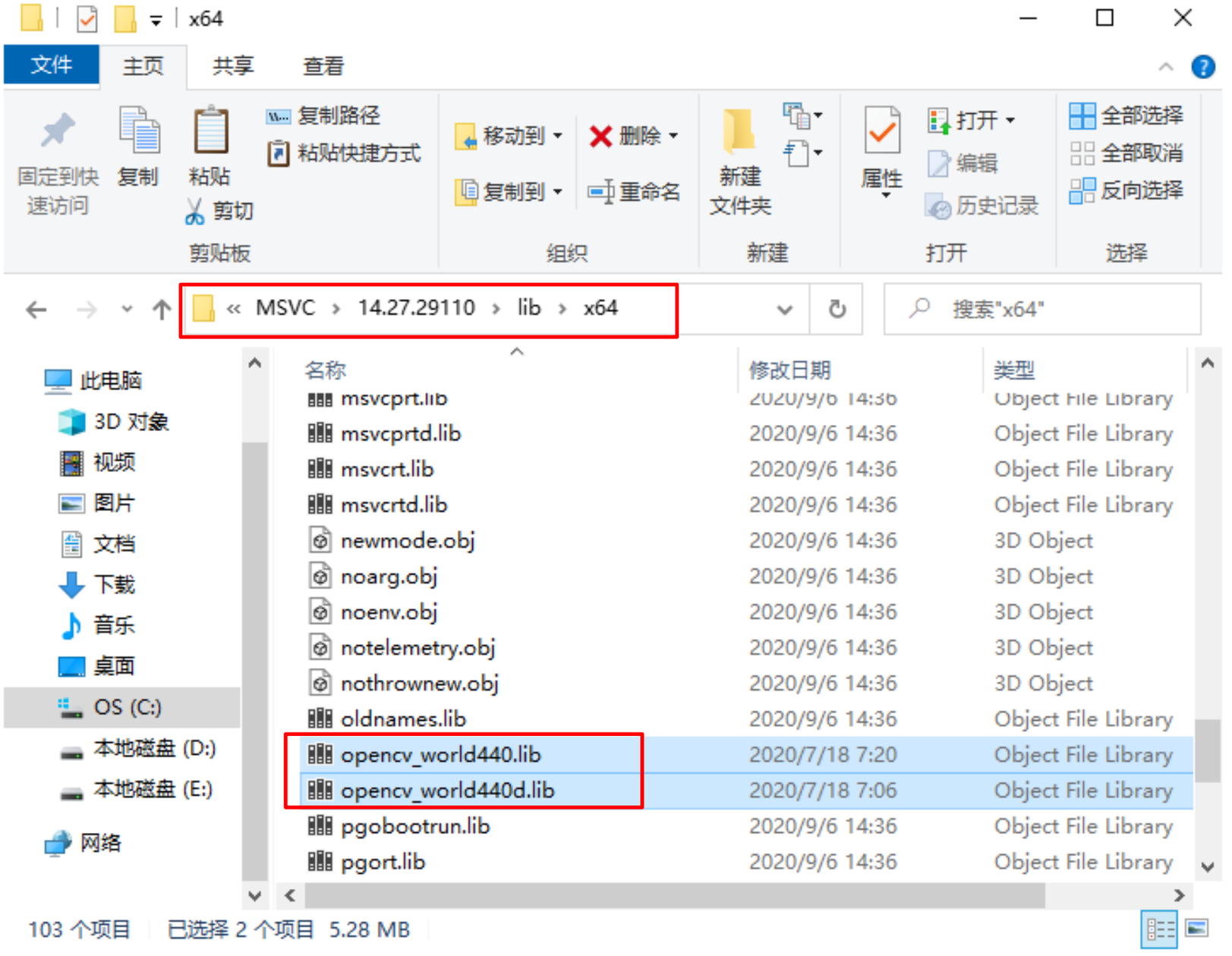


Open the “Properties” window for the second time, and choose “Library Directories”

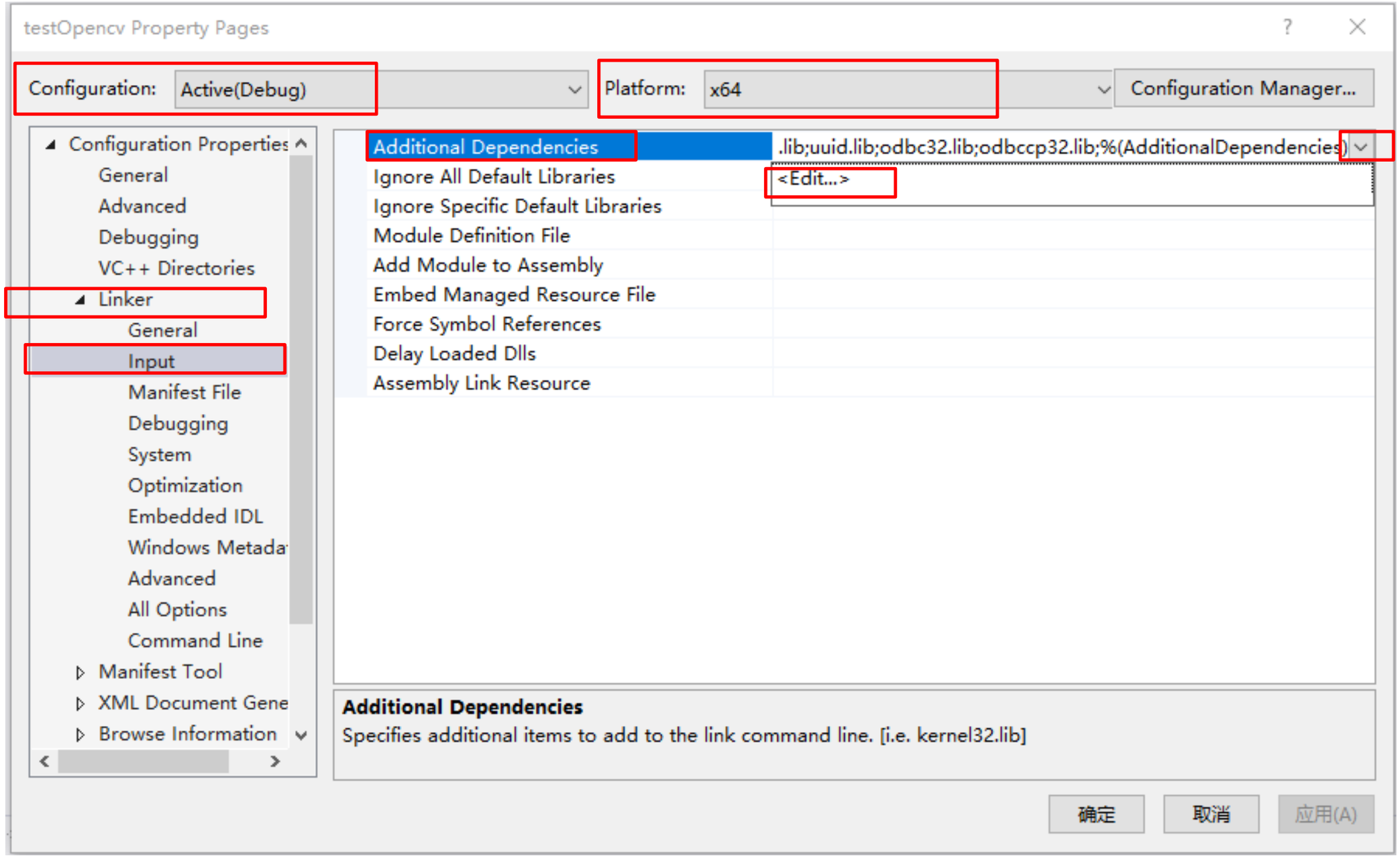



Library Directories
Evaluated value:
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.27.29110\lib\x64
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.27.29110\atlmfc\lib\x64
C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\VS\lib\x64
Inherited values:
$(VC_LibraryPath_x64)
$(WindowsSDK_LibraryPath_x64)
Inherit from parent or project defaults
Macros>>
OK
Cancel
copy the directory

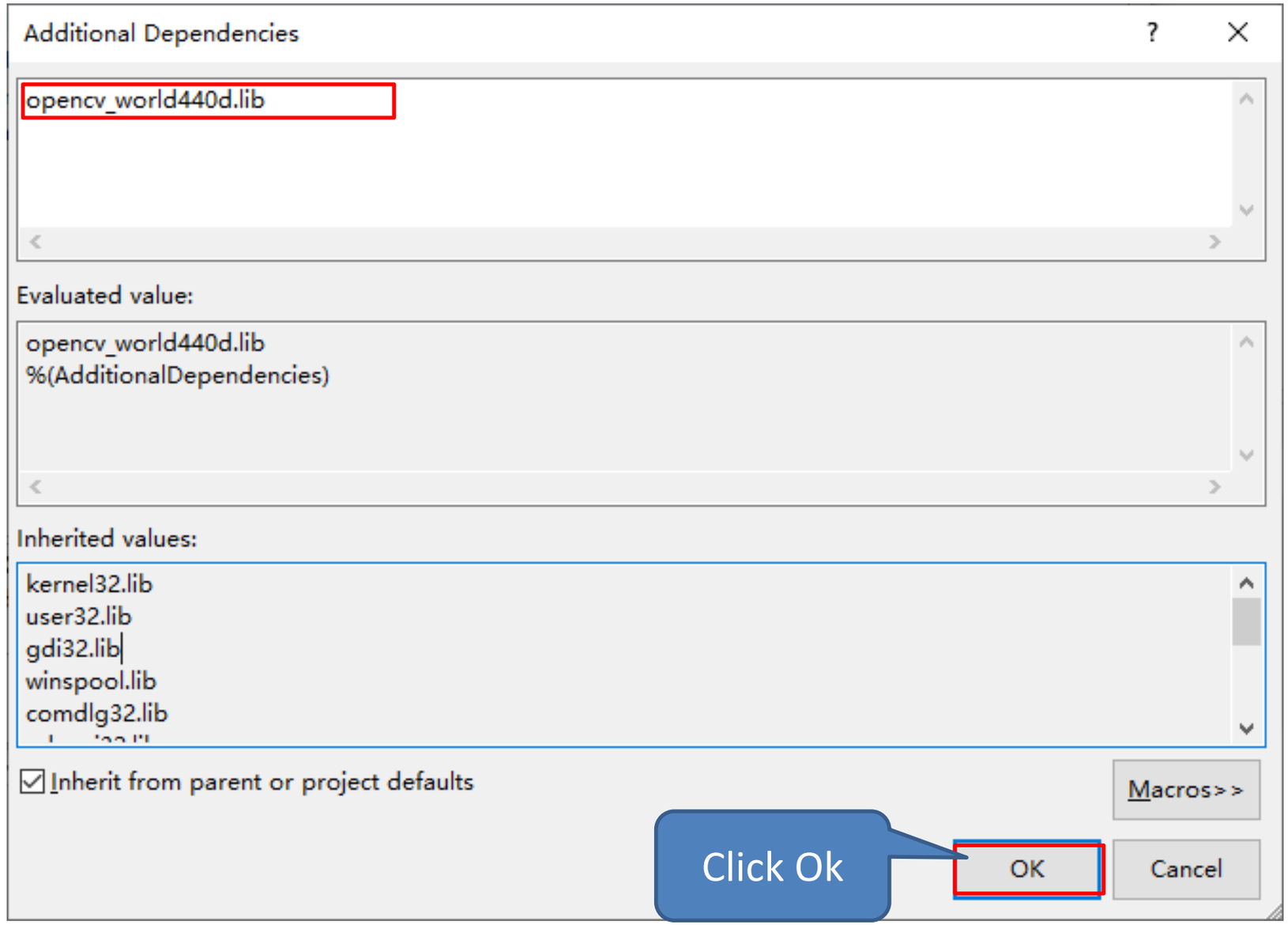
Copy the two files(opencv_world440.lib and opencv_world440d.lib) which are in the “D:\Software\opencv\build\x64\vc15\lib” to the directory you just copied

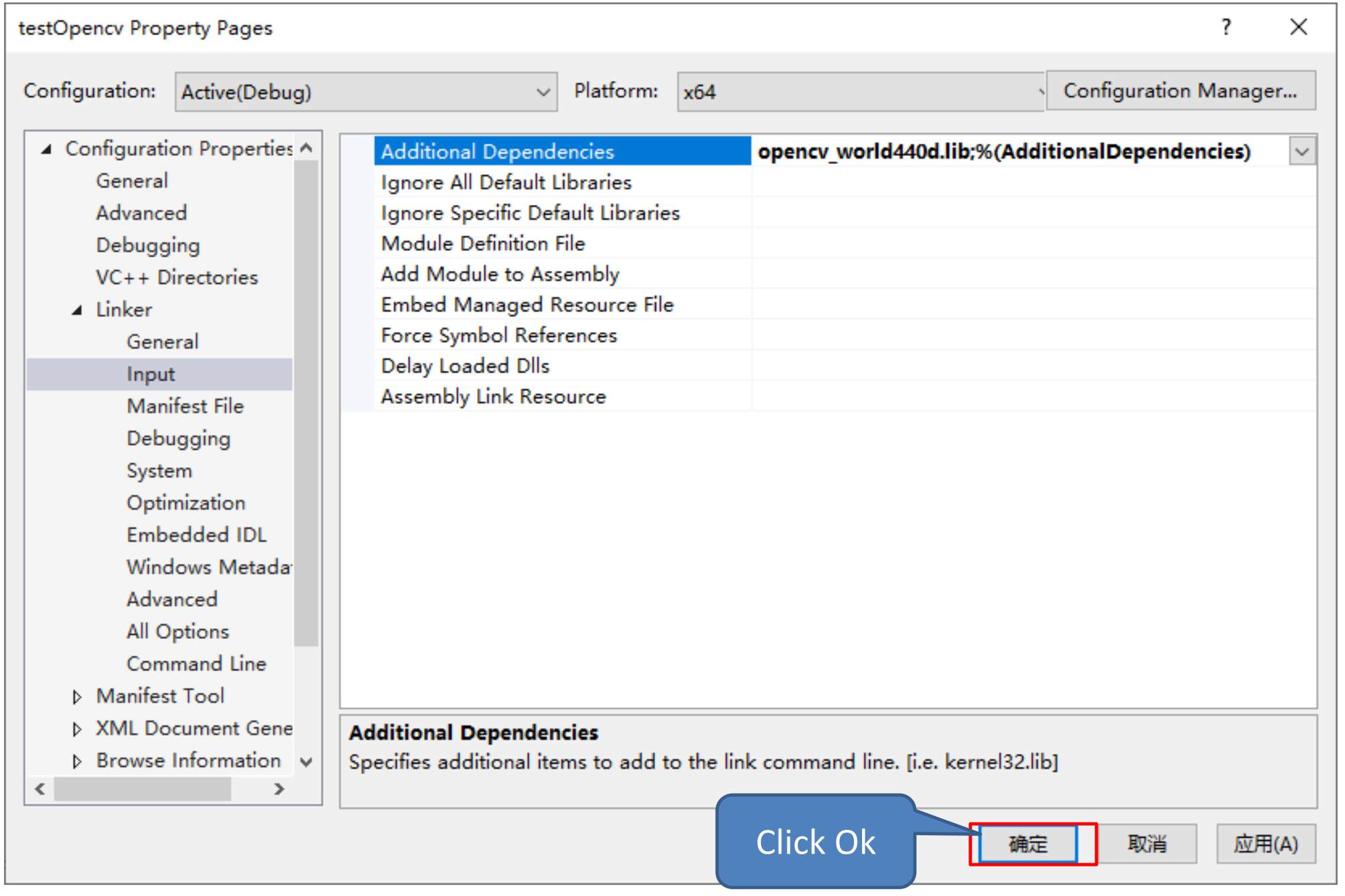


Open the “Properties” window for the third time, and choose “Linker”



Copy the filename “opencv_world440d.lib” in the D:\Software\opencv\build\x64\vc15\lib folder to the “Additional Dependencies”




testOpencv Property Pages
Configuration: Active(Debug)
Platform: x64
Configuration Manager...
Configuration Properties
General
Advanced
Debugging
VC++ Directories
Linker
General
Input
Manifest File
Debugging
System
Optimization
Embedded IDL
Windows Metada
Advanced
All Options
Command Line
Manifest Tool
XML Document Gene
Browse Information
Additional Dependencies
opencv_world440d.lib;%(AdditionalDependencies)
Ignore All Default Libraries
Ignore Specific Default Libraries
Module Definition File
Add Module to Assembly
Embed Managed Resource File
Force Symbol References
Delay Loaded Dlls
Assembly Link Resource
Additional Dependencies
Specifies additional items to add to the link command line. [i.e. kernel32.lib]
Click Ok
确定
取消
应用(A)

Test your opencv

Notice: Debug must be x64

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) testOpencv

Debug x64 Local Windows Debugger Live Share

testOpencv.cpp

testOpencv (Global Scope)

```
#include <opencv2/opencv.hpp>
using namespace cv;

int main()
{
    Mat image = imread("1.jpg");  // read a picture
    imshow("我的图片", image);    // show the picture

    waitKey(0);        // wait

    return 0;

}
```

100 % No issues found Ln: 14 Ch: 1 TABS CRLF

Output

Show output from: Debug

```
The thread 0x3e60 has exited with code 0 (0x0).
The thread 0x50e8 has exited with code 0 (0x0).
The thread 0x2d68 has exited with code 0 (0x0).
The thread 0x4798 has exited with code 0 (0x0).
The thread 0x1b0 has exited with code 0 (0x0).
The program '[20784] testOpencv.exe' has exited with code 0 (0x0).
```

Error List Output

Solution Explorer

Search Solution Explorer (Ctrl+;)

Solution 'testOpencv' (1 of 1 project)

testOpencv

References

External Dependencies

Header Files

Resource Files

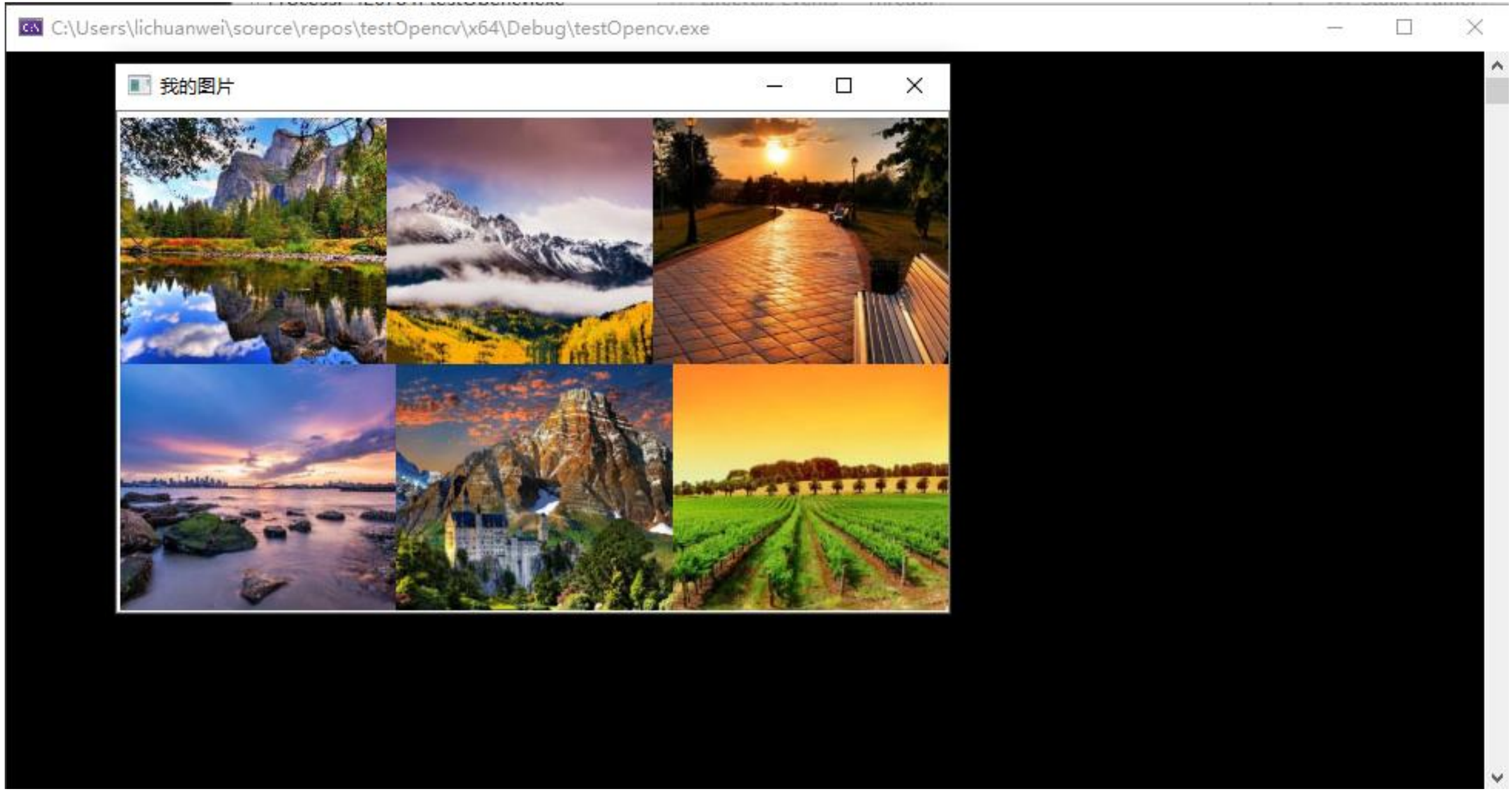
Source Files

testOpencv.cpp

Solution Explorer Team Explorer

Properties

Ready Add to Source Control


C:\Users\lichuanwei\source\repos\testOpencv\x64\Debug\testOpencv.exe
我的图片

Install openCV on macOs

https://blog.csdn.net/wo164683812/article/details/80114999

https://www.cnblogs.com/linjk/p/6029306.html

https://github.com/poloclub/cnn-explainer

README.md

CNN Explainer

An interactive visualization system designed to help non-experts learn about Convolutional Neural Networks (CNNs)

Build Status arxiv badge DOI:10.1109/TVCG.2020.3030418

For more information, check out our manuscript:

CNN Explainer: Learning Convolutional Neural Networks with Interactive Visualization. Wang, Zijie J., Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng Chau. *IEEE Transactions on Visualization and Computer Graphics (TVCG), 2020.*

Live Demo

For a live demo, visit: http://poloclub.github.io/cnn-explainer/

3 Exercises

Write a function **calculateAverage()** which takes four int arguments which are marks for four courses in the semester and returns their average as a float.

The **calculateAverage()** function should take only valid range for marks which is between 0-100. If the marks are out of range throw an **OutOfRangeException** – define this exception as a class.

Invoke the **calculateAverage()** function in main function and get the following inputs and outputs:

```
Please enter marks for 4 courses:70 80 90 67
The average of the four courses is 76.75
Would you want to enter another marks for 4 courses(y/n)?y
Please enter marks for 4 courses:120 56 89 99
The parameter 1 is 120 which out of range(0-100).
Would you want to enter another marks for 4 courses(y/n)?y
Please enter marks for 4 courses:90 -87 67 92
The parameter 2 is -87 which out of range(0-100).
Would you want to enter another marks for 4 courses(y/n)?n
Bye, see you next time.
```