# Environment

OS: Windows 7 Ultimate x64

Compiler: Visual Studio 2010

Boost: 1.57.0 (installation path is “D:\libs\boost”)

CMake: 3.2.1

Python3: 3.3.5

NumPy: 1.9.1

SciPy: 0.14.0

PyQt4: 4.10.3

Pyparsing: 2.0.1

python-dateutil: 2.2

six: 1.5.2

Git: 1.9.5

# Downloading

Boost: <http://sourceforge.net/projects/boost/files/boost/1.57.0/>

CMake: <http://www.cmake.org/files/v3.1/cmake-3.1.3-win32-x86.exe>

Python3: <http://www.python.org/ftp/python/3.3.5/python-3.3.5.msi>

NumPy: <http://sourceforge.net/projects/numpy/files/NumPy/1.9.1/numpy-1.9.1-win32-superpack-python3.3.exe/download>

SciPy: <http://sourceforge.net/projects/scipy/files/scipy/0.14.0/scipy-0.14.0-win32-superpack-python3.3.exe/download>

PyQt4: <http://sourceforge.net/projects/pyqt/files/PyQt4/PyQt-4.10.3/PyQt4-4.10.3-gpl-Py3.3-Qt4.8.5-x32.exe/download>

Pyparsing: <http://sourceforge.net/projects/pyparsing/files/pyparsing/pyparsing-2.0.1/pyparsing-2.0.1.win32-py3.3.exe/download>

setuptools: <https://pypi.python.org/pypi/setuptools#downloads>

python-dateutil: <https://pypi.python.org/pypi/python-dateutil/2.2>

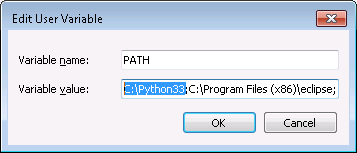
six: <https://pypi.python.org/pypi/six/1.5.2>

Matplotlib: <http://sourceforge.net/projects/matplotlib/files/matplotlib/matplotlib-1.3.1/matplotlib-1.3.1.win32-py3.3.exe/download>

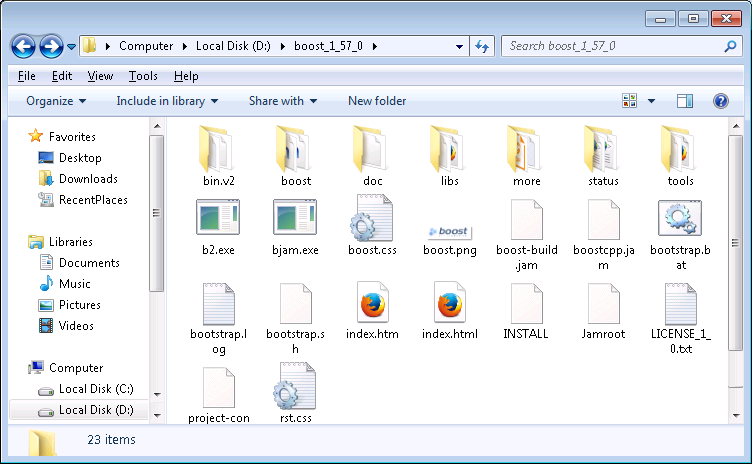
Git: <https://github.com/msysgit/msysgit/releases/download/Git-1.9.5-preview20141217/Git-1.9.5-preview20141217.exe>

# Compiling Boost

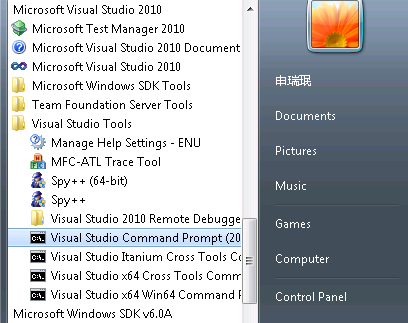
* Installing Python 3, and make sure its path is included in the PATH environment variable:



* Extracting source code files of Boost:



* Start the x86 command prompt of Visual Studio 2010:



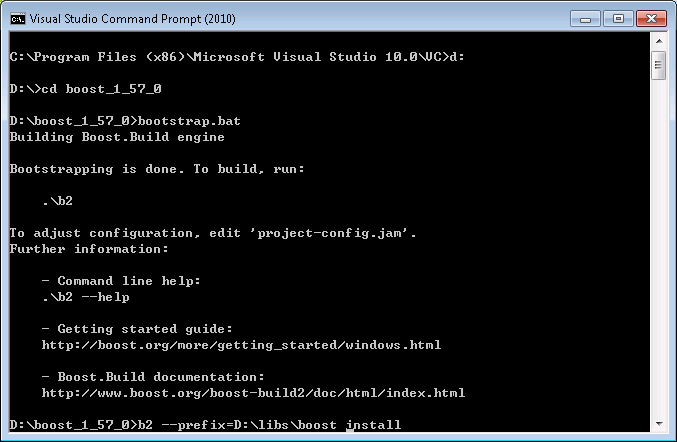
* Compiling Boost (both static and shared link):

d:

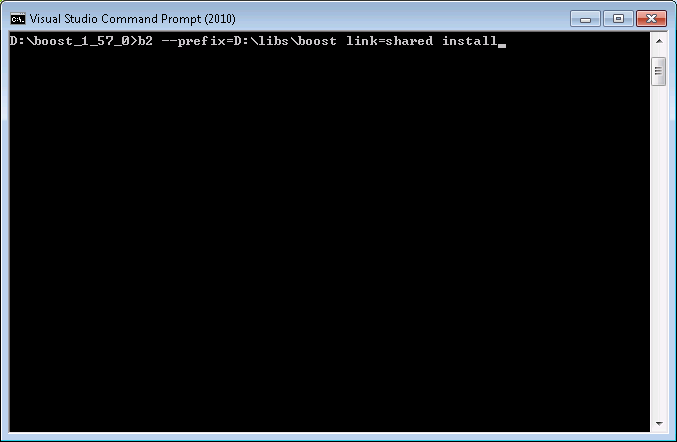
cd D:\boost\_1\_57\_0

bootstrap.bat

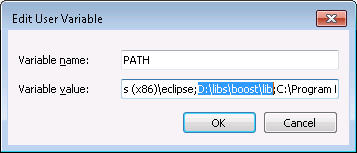
b2 --prefix=D:\libs\boost install

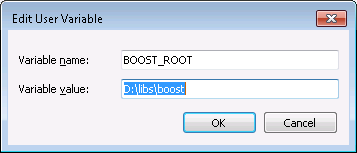


b2 --prefix=D:\libs\boost link=shared install

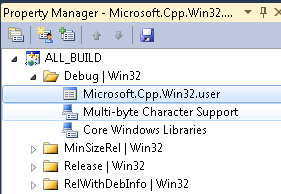


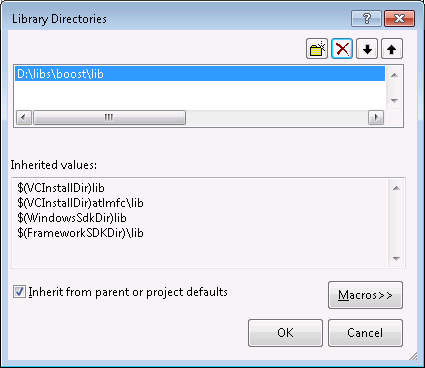
* Adding the installation path of Boost “D:\libs\boost” into both the PATH and BOOST\_ROOT environment variables:





* Open the property manager of Visual Studio 2010, and adding the library path of Boost “D:\libs\boost\lib” into library directories:



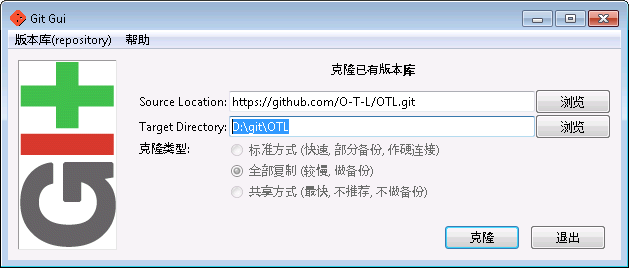


# Configuring OTL

* Clone the repository:

Clone URL: <https://github.com/O-T-L/OTL.git>

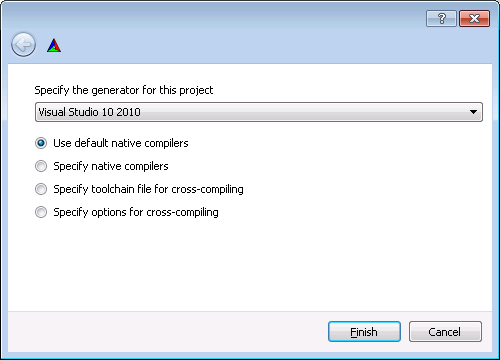
Destination: D:\git\OTL

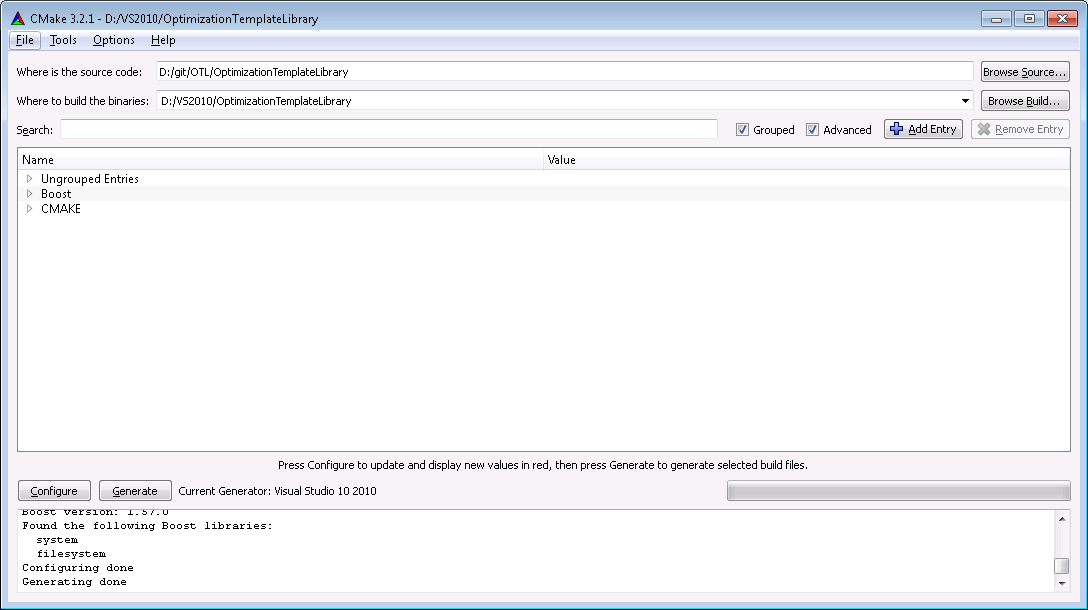


* Generating project files of Visual Studio 2010

Source code directory: D:\git\OTL\OptimizationTemplateLibrary

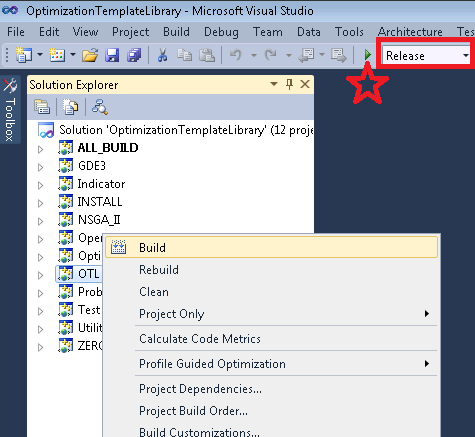
Binary directory: D:\VS2010\OptimizationTemplateLibrary



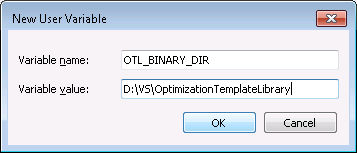


* Building the OTL project:

Choosing the “Release” building type will be better.



* Adding the binary directory of OTL “D:\VS2010\OptimizationTemplateLibrary” into the OTL\_BINARY\_DIR environment variable (optional):



# Configuring PyOTL

* Clone the repository:

Clone URL: <https://github.com/O-T-L/PyOTL.git>

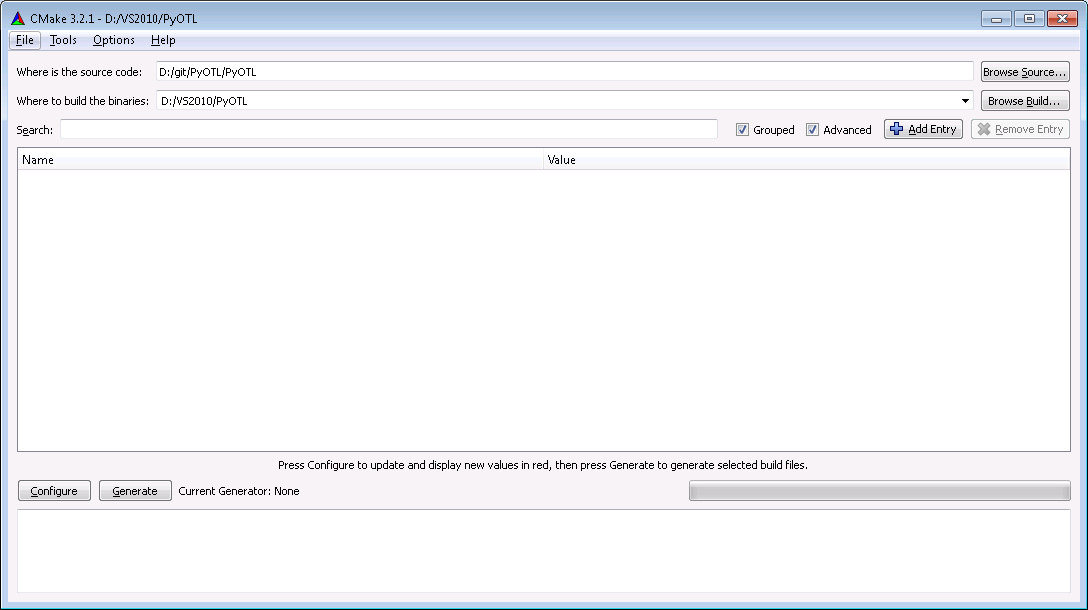
Destination: D:\git\PyOTL



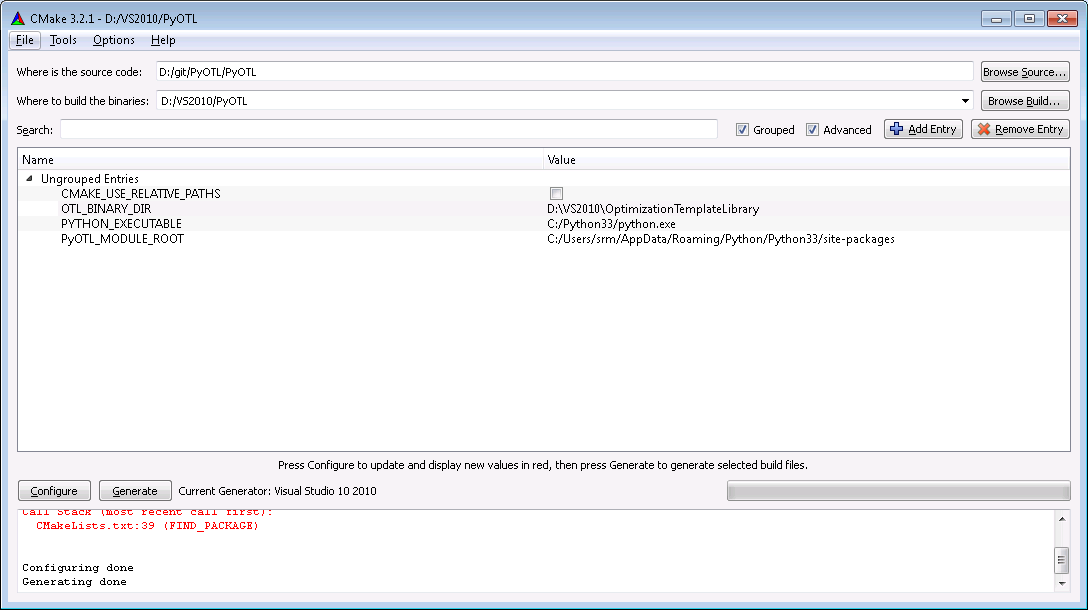
* Generating project files of Visual Studio 2010

Source code directory: D:\git\PyOTL\PyOTL

Binary directory: D:\VS2010\PyOTL

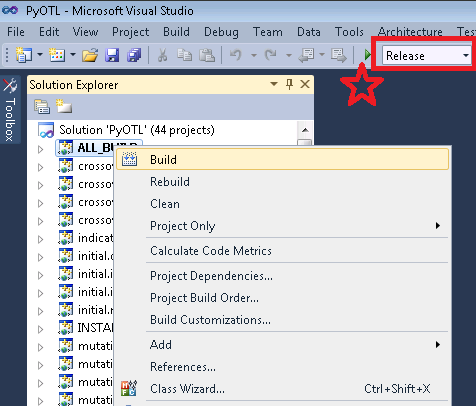


* If the OTL\_BINARY\_DIR environment variable is not set, then set the OTL\_BINARY\_DIR CMake variable into “D:\VS2010\OptimizationTemplateLibrary”:

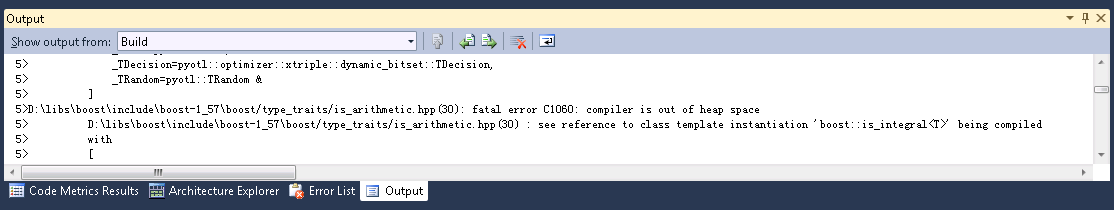


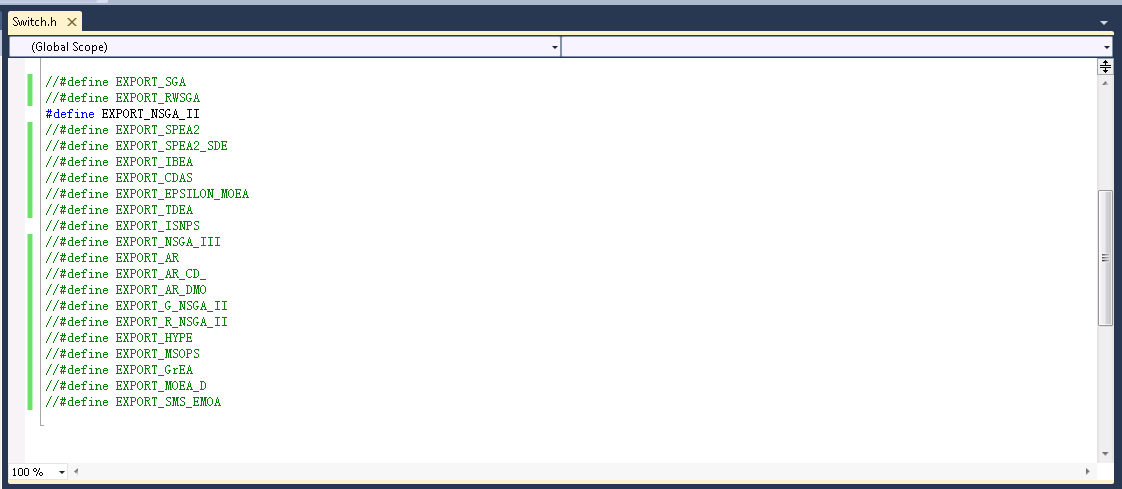
* Compiling all projects of PyOTL:

Choosing the “Release” building type will be better.



* If your memory is not enough, you can disable some optimization algorithms in “D:\git\PyOTL\PyOTL\Include\pyotl\optimizer\Switch.h”:





# Configuring PyOptimization

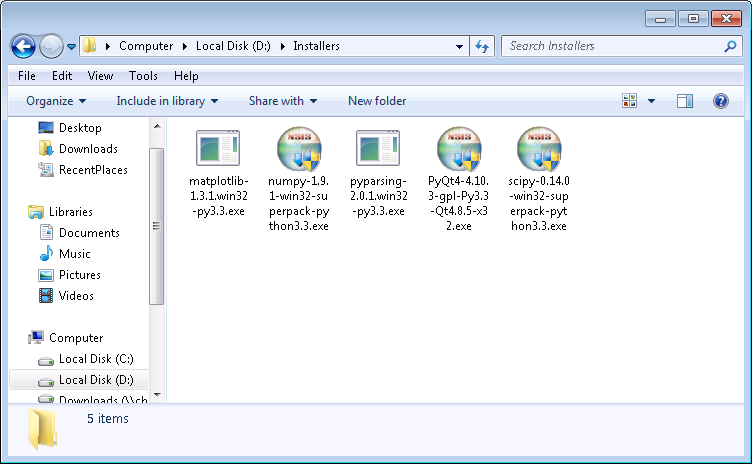
* Clone the repository:

Clone URL: <https://github.com/O-T-L/PyOptimization.git>

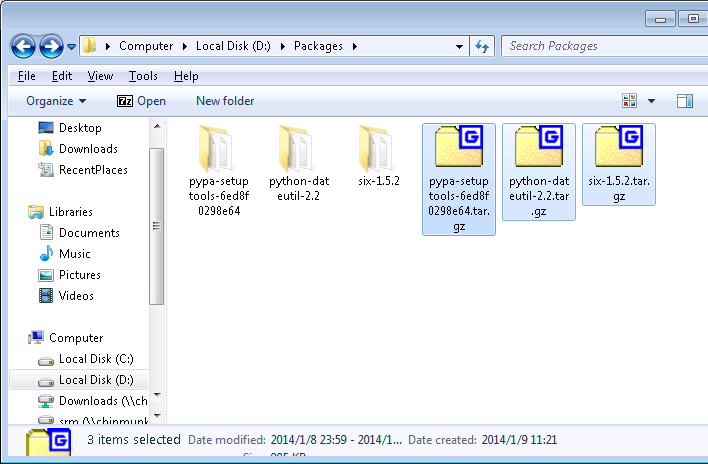
Destination: D:\git\PyOptimization



* Installing the Python libraries depended by PyOptimization:



* Manually compiling the Python libraries depended by PyOptimization:

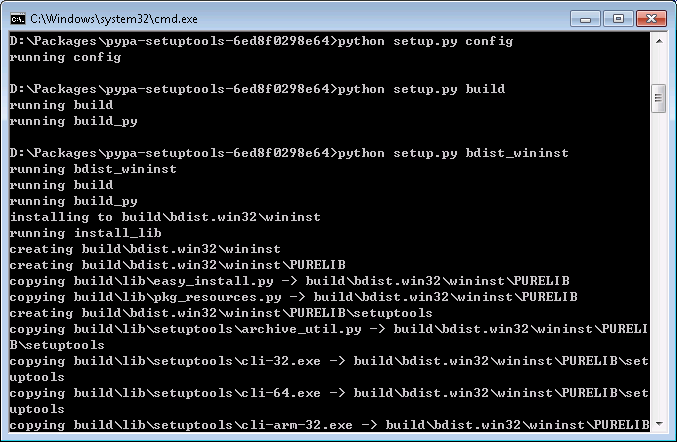


* Compiling setuptools:

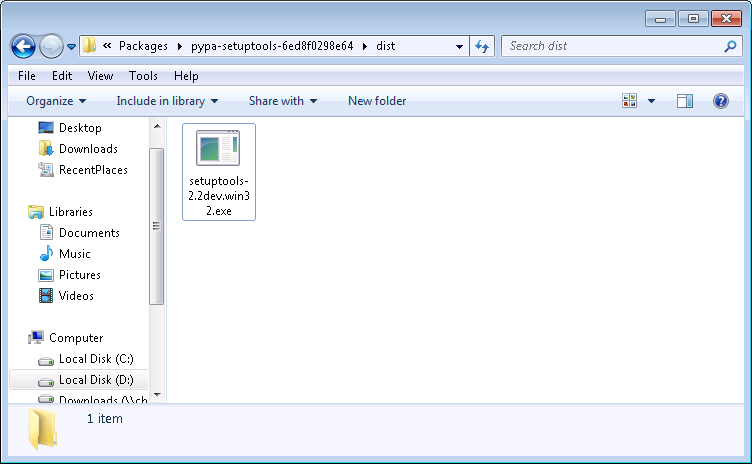
python setup.py config

python setup.py build

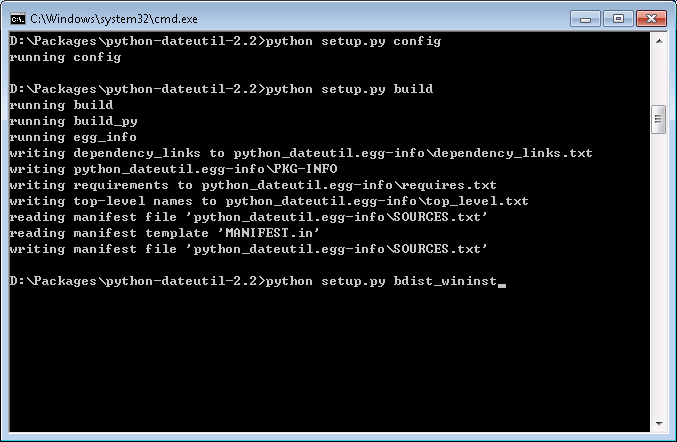
python setup.py bdist\_wininst

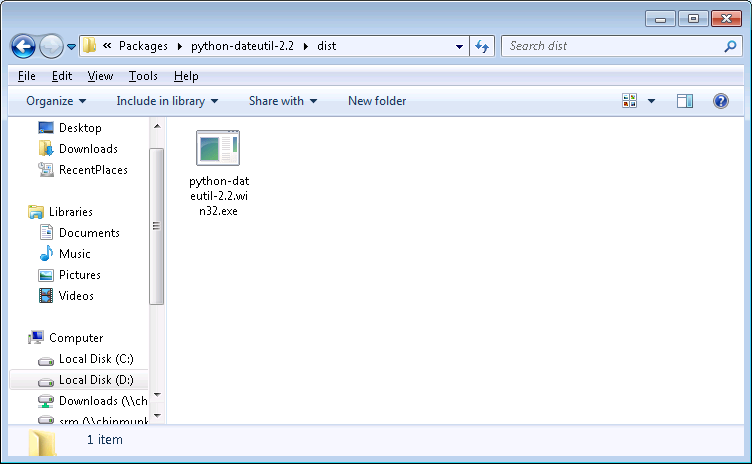


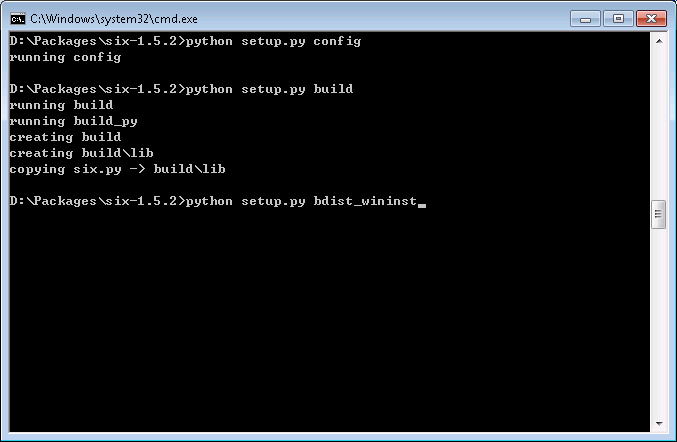
Run the installer as administrator.

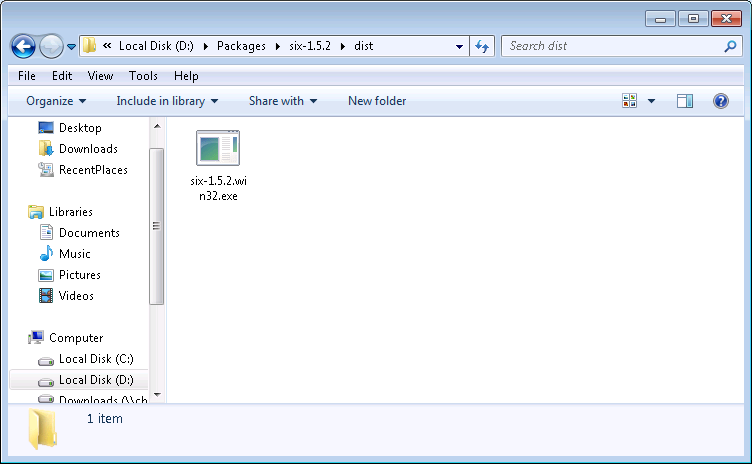


* Use the same way to compile python-dateutil and six:









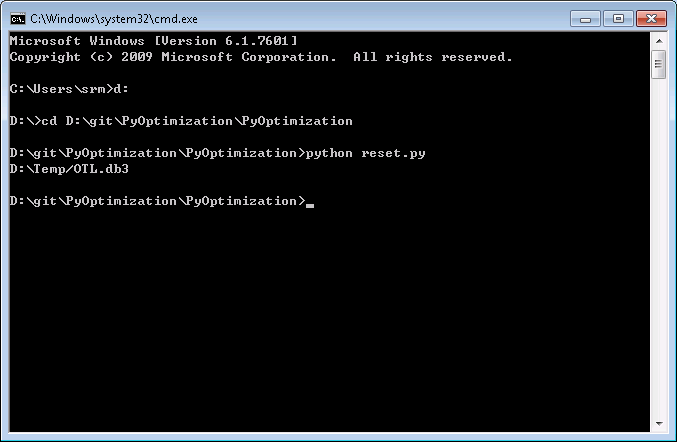
# Using PyOptimization

* Creating an empty database:

d:

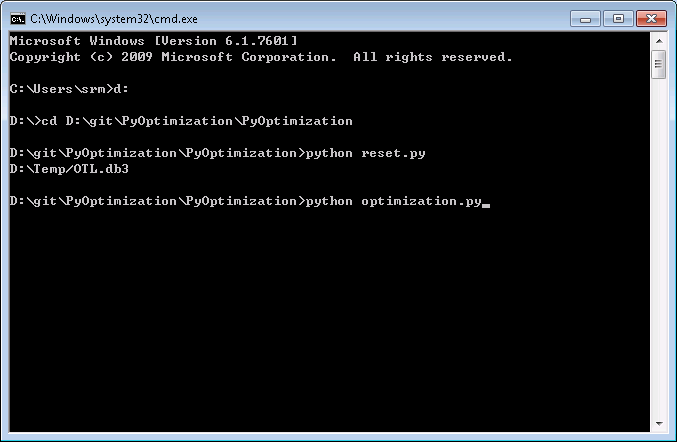
cd D:\git\PyOptimization\PyOptimization

python reset.py



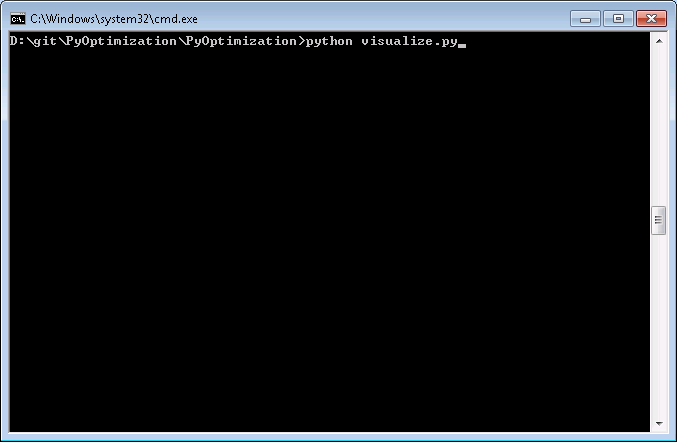
* Running optimization algorithms:

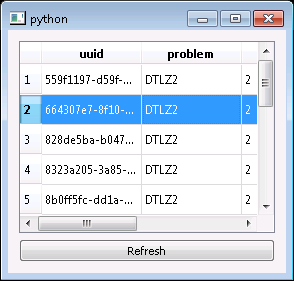
python optimization.py

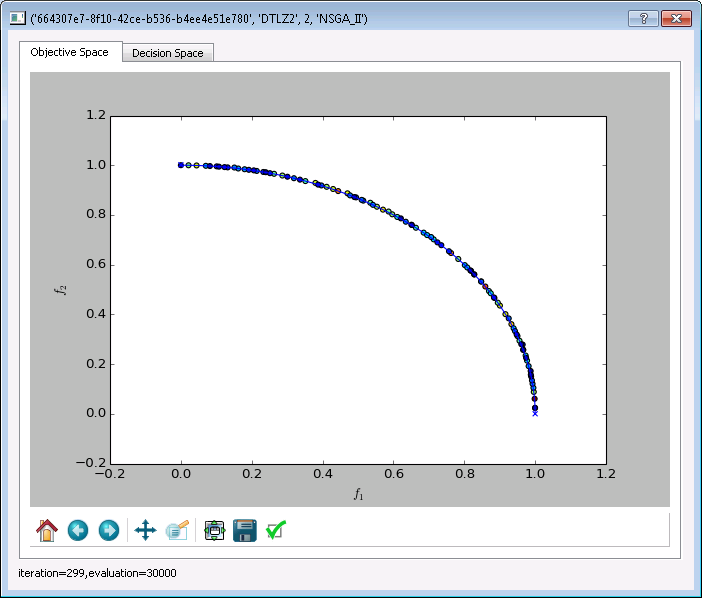


* Visualizing the results:

python visualize.py







* Evaluating the results:

python visualize.py

