1 TABLE OF CONTENTS

2 WINDOWS CPLEX COMPILATION – INFO & INSTRUCTIONS

To execute the code, we need in order:

- Visual Studio IDE (note: this is different from Visual Studio Code)
  - Installing the Community Version from here
- A C++ compiler
  - You can either use MinGW (here) or MSVC (done when selecting "Develop C++ applications" when installing Visual Studio
- CPLEX Studio installed on your machine (current version is 22.11)
  - All of the info present in the Moodle of the course here

The problem on Windows is evidenced by the fact that fatal errors might occur, like:

```
lex/ilocplex.h: No such file or directory
 #include <ilcplex/ilocplex.h>
          ^~~~~~~~~~~~~~~~~~~~
compilation terminated.

Build finished with error(s).
The terminal process failed to launch (exit code: -1).

Terminal will be reused by tasks, press any key to close it.
```

As seen here, a normal execution of Cplex would include:

`C:\Program Files\IBM\ILOG\CPLEX_Studio_Community201\cplex\include`

`C:\Program Files\IBM\ILOG\CPLEX_Studio_Community201\concert\include`

These folder need to be configured inside of the additional inclusions and also additional dependencies in the form of files and directives to the compiler.

The most recent versions (up to 2019, current is 2022) do not allow complete editing of the compilation options as you might see here.

As found within the internal group of the course, the main problem seems to be that there is not `cpxmacro.h` file inside of Windows installations, so if we try to use it in our project we have problem with functions for declaration of env, adding variables and constraints.

3 WINDOWS CPLEX COMPILATION – SOLUTION 1

The solution is actually the following:

- Once you have installed Visual Studio and CPLEX on your machine, you should take some ready-made examples, so to import a Solution file (basically, a configuration file which needs to be imported in order to make the execution work) and then an executable file with a main()

The paths to consider for solution files are the following:

- C++ files:

```
C:\Program
Files\IBM\ILOG\CPLEX_Studio2211\cplex\examples\x64_windows_msvc14\stat_mda
```

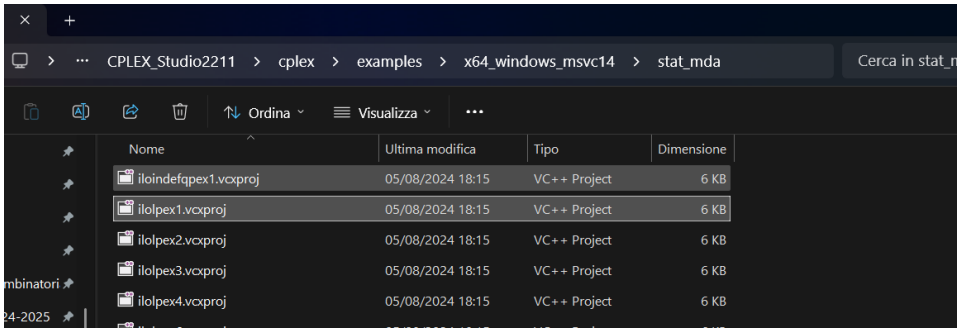
- C files:

```
C:\Program
Files\IBM\ILOG\CPLEX_Studio2211\cplex\examples\x64_windows_msvc14\stat_mdd
```

These folders report a lot of different files which are the "Solution" files; we need to consider files with extension `.vcxproj`. The goal here would be to first select a Solution file and then select a C/C++ file of the same name. So:

- If you want to execute a C example, go the "mda"
- If you want to execute a C++ example, go to "mdd"

For instance, let's consider `ilolpex1.vcproj`, which is a C++ file:



We then need the actual source codes, which are to be linked with the respective `vcxproj` files of before. Once again, it's different for both formats:

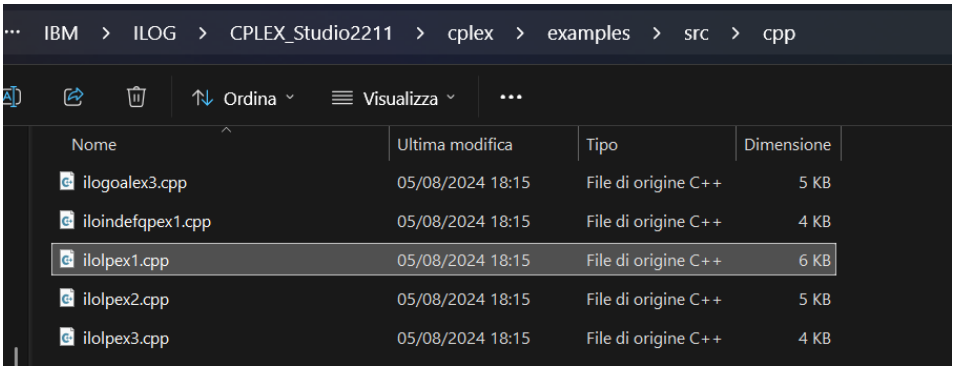
- C files:

```
C:\Program Files\IBM\ILOG\CPLEX_Studio2211\cplex\examples\src\c
```

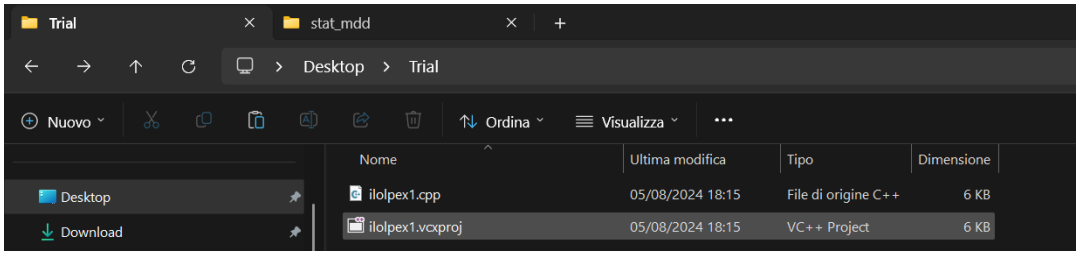
- C++ files:

```
C:\Program Files\IBM\ILOG\CPLEX_Studio2211\cplex\examples\src\cpp
```

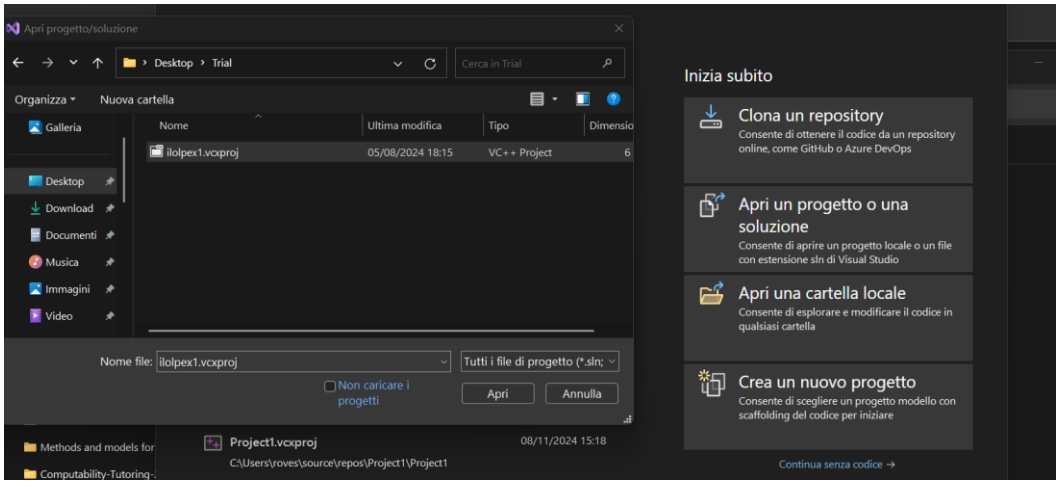
We then take the file of the same name as before:



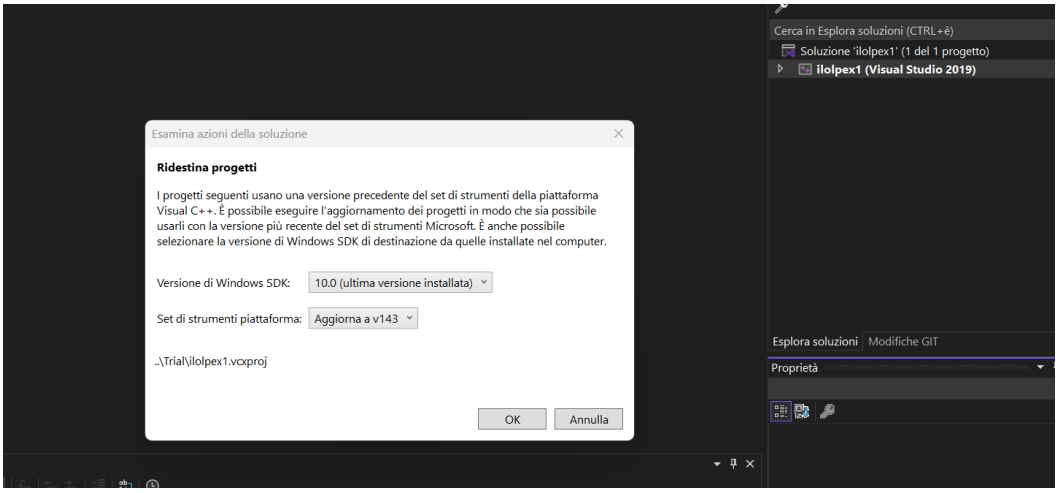
We need both files in order to import them into Visual Studio and then customize the code of the actual source file (C/C++) so to make the code work fine. We then create a folder with a custom name on a custom location with both files, like the following:



We then open Visual Studio loading the vcxproj file on “Open a project or a solution” file.

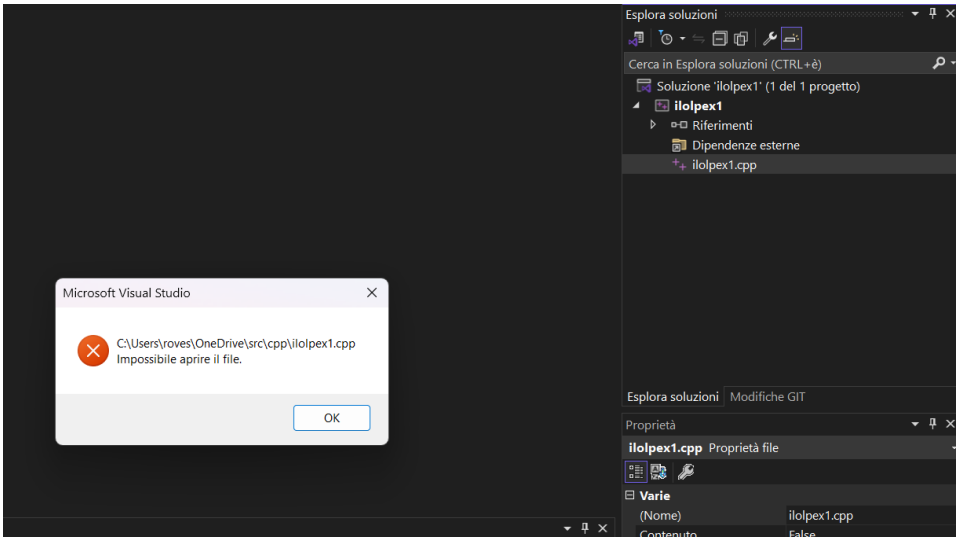


Once the prompt is open, select the Windows SDK version and multiplatform by default and continue.

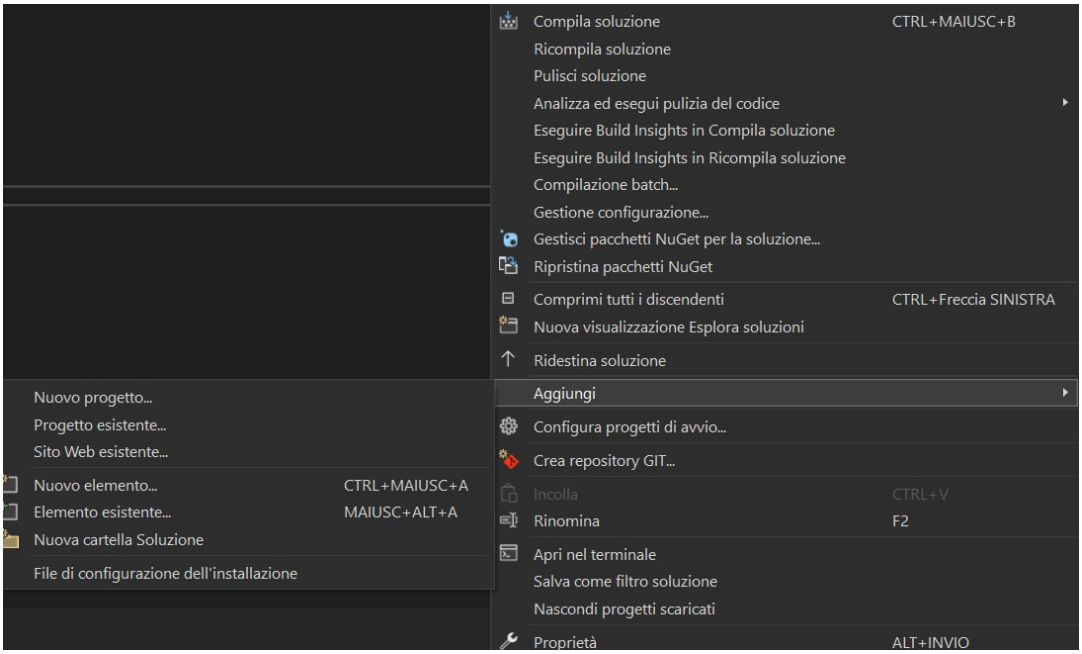


WARNING

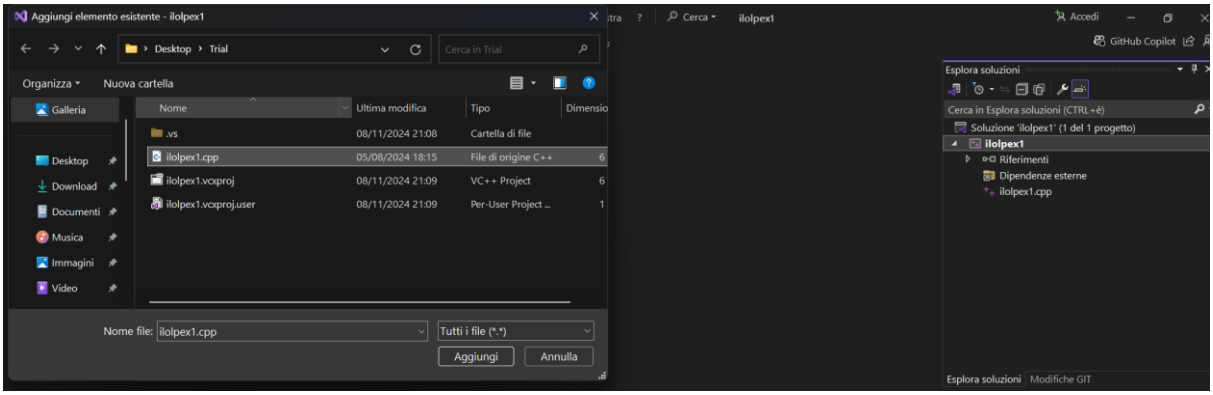
At this point, since the vcxproj file points to files present in the previous path (so inside of the Cplex path), it will tell you “Impossible to open file”, since it does not see the local path:



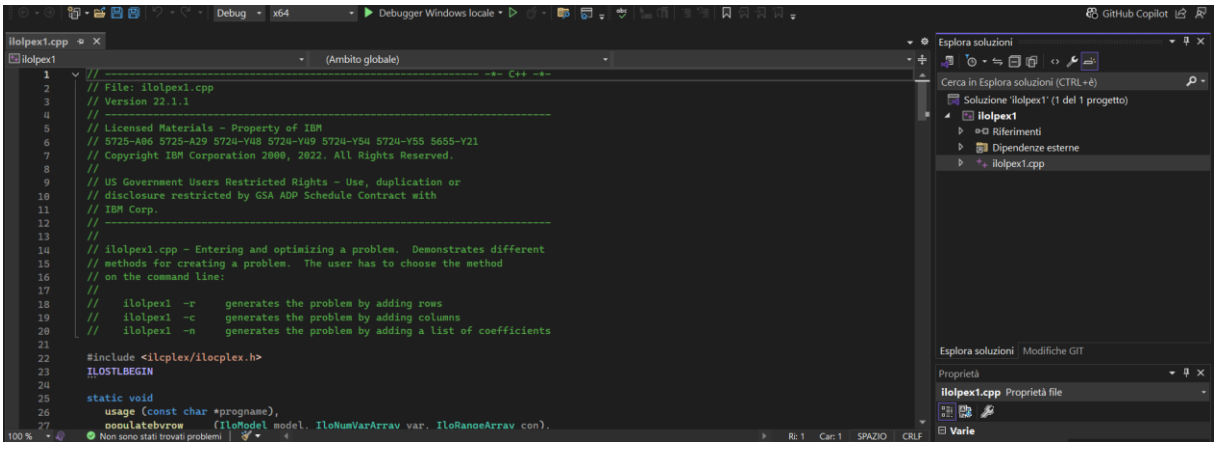
What you will do to solve this problem, is to right-click the name of project in the right menu present (in this case where there is `ilolpex1`) and then click “Add” (Aggiungi) and then click on “Existing element” (Elemento esistente):



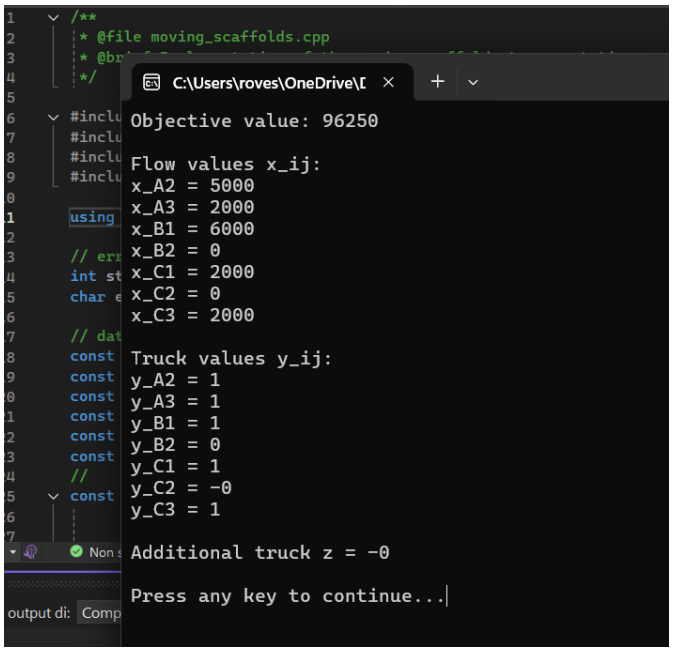
Here we will select the actual C/C++ file:



Please remove the old file which is not to be found, so you have only one, the correctly imported file. You should see something like this:



We then build the actual file, and a command prompt window will feedback the right execution here (this is a different execution, the example run in the laboratory, so you have an idea):

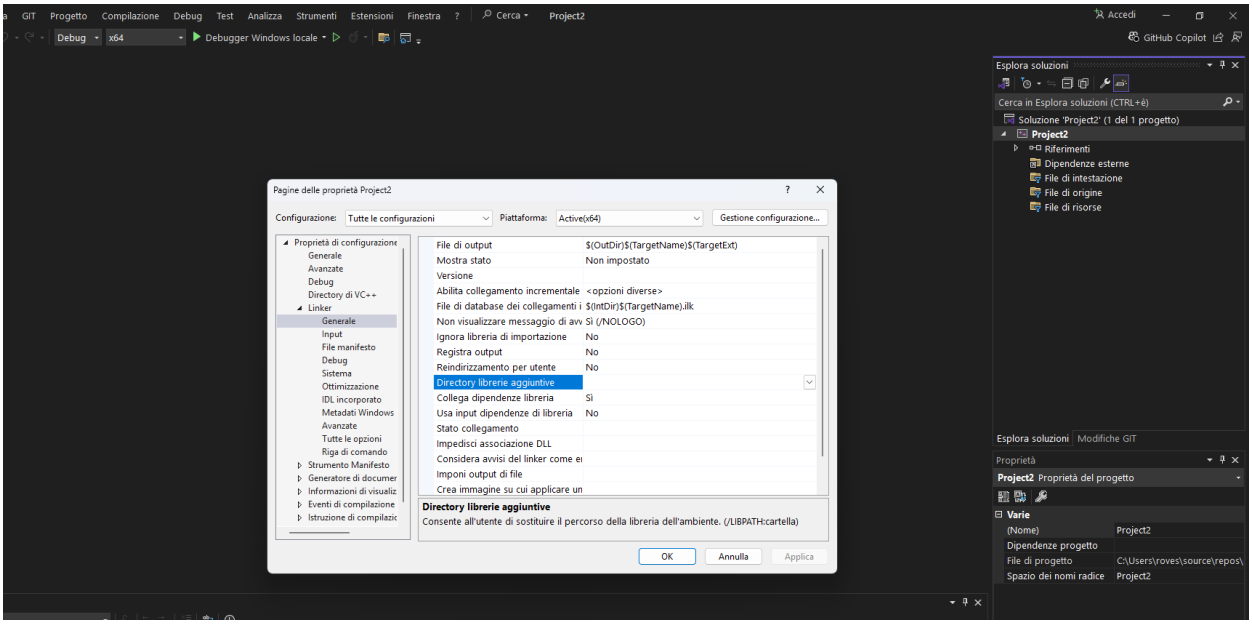


This way, any kind of project works. This was tested both on C and C++ files.

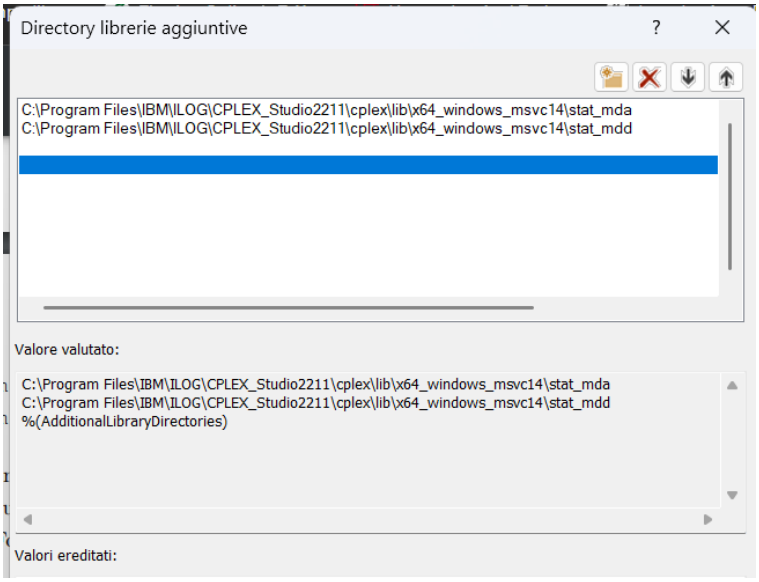
4 WINDOWS CPLEX COMPILATION – SOLUTION 2

Another way to make this work is to create a C++ project from scratch and then right click on the right side menu on Properties so to open the following window – adapted from 4-5 page of this.

Then, one goes to “Linker” > “General” > “Additional library directories”:



Here, one then adds the mda/mdd folders as path:

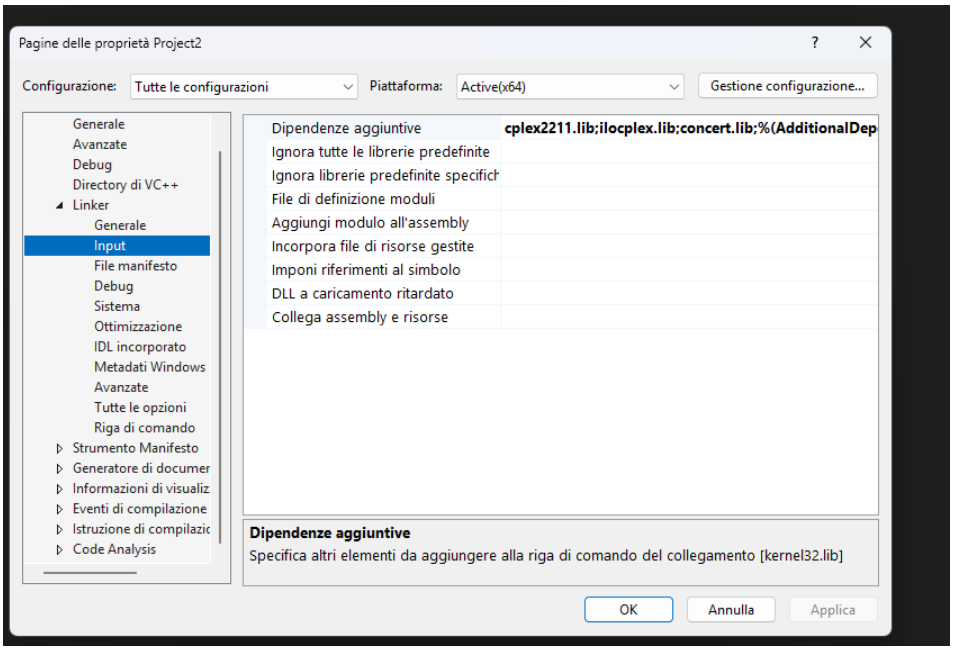


which are:, I remember:

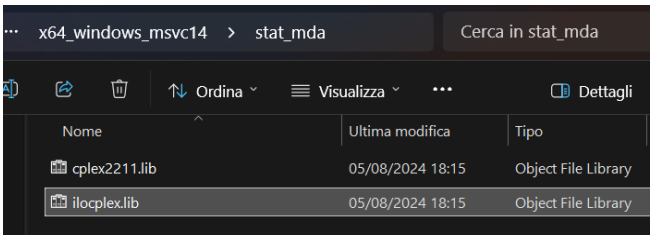
```
C:\Program Files\IBM\ILOG\CPLEX_Studio2211\cplex\lib\x64_windows_msvc14\stat_mdd
```

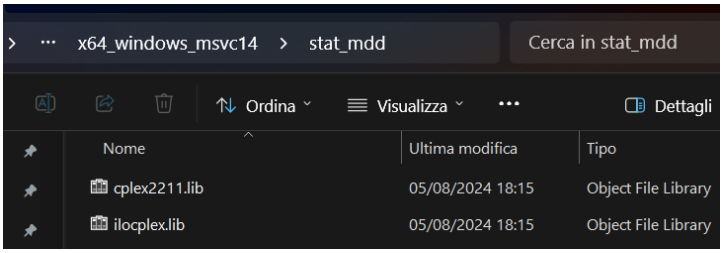
```
C:\Program Files\IBM\ILOG\CPLEX_Studio2211\cplex\lib\x64_windows_msvc14\stat_mda
```

Then, in the “Linker” > “Input” tab, click on “Additional dependencies”:



Add all of the files which are `.lib` files inside of the mda/mdd folders:





They are `concert.lib` (concert directory of before and the two above files), separated with a semicolon when inserted: