Detalii legate de laborator

laborator API

-20% sau 40% din nota finala

prima parte: MPI

-folosim visual studio, MPI e o librarie

-4 laboratoare legate de MPI

-avem de facut probleme pe care le prezentam la laboratorul urmator (o saptamana de introducere in concept si dupa o saptamana in care prezentam)

a doua parte: thread-uri in C++ si STL

-ne alegem o tema dintr-o lista pe care o implementam cu MPI, thread-uri si STL paralel

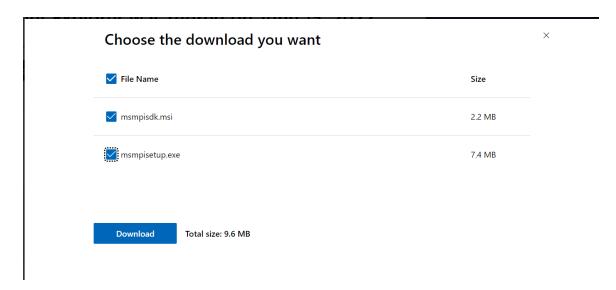
-daca vrem sa ne marim nota, putem sa prezentam mai multe laboratoare intr-o saptamana

-stabilim la inceput profesorul la care venim, nu putem sa venim la amandoi

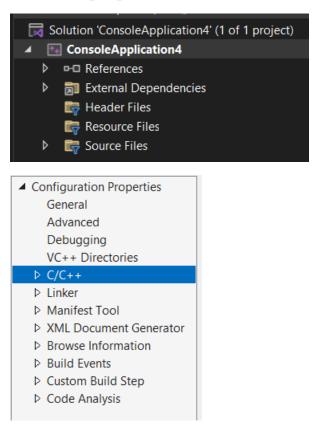
Cum se face legatura intre libraria MPI si Visual Studio

-descarcam libraria MPI

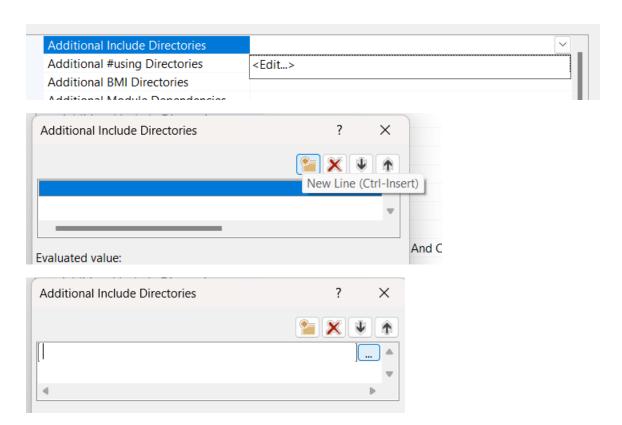
https://www.microsoft.com/en-us/download/details.aspx?id=105289



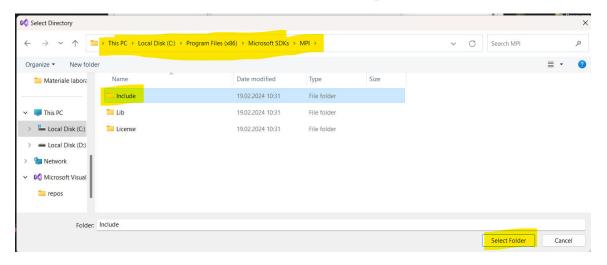
- -dupa ce am descarcat, intram in downloads si le rulam pe amandoua
- -cream un console application C++ in Visual Studio
- -click dreapta pe NUMELE PROIECTULUI, apoi Properties



-apasam pe Edit..., apoi New Line, apoi ...

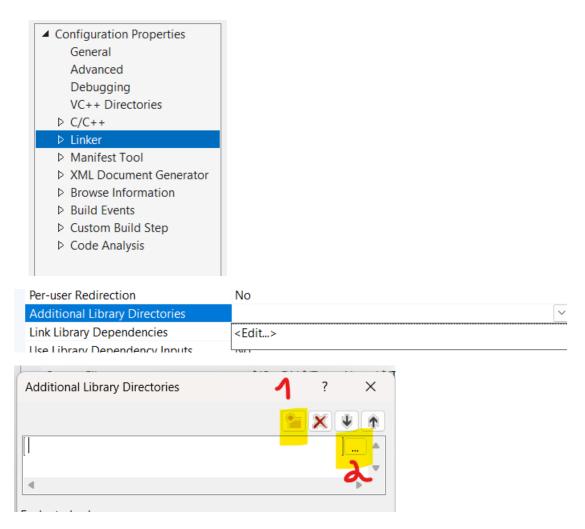


-selectam de la locatia aceasta fisierul include, apoi Select folder

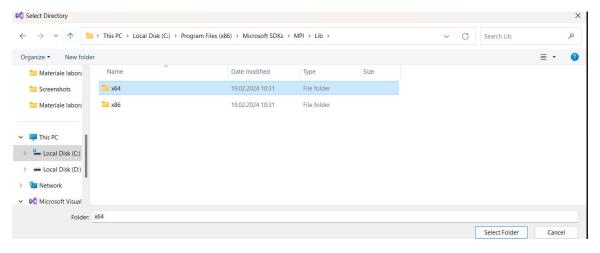


-apoi dam ok

-tot la Properties selectam si Linker, dam pe Edit... la Additional Library Directions, apoi New Line si ...



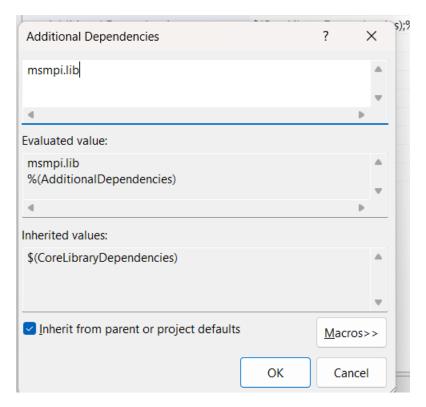
-selectam folderul de la locatia aleasa



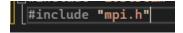
-tot in Properties ne ducem aici

▲ Configuration Properties General Advanced Debugging VC++ Directories ▷ C/C++ ▲ Linker General Input Manifest File Debugging System Optimization Embedded IDL Windows Metadata Advanced All Options Command Line ▶ Manifest Tool > XML Document Generato ▶ Browse Information ▶ Build Events ▶ Custom Build Step

Additional Dependencies	\$(CoreLibraryDependencies);%(AdditionalDependencies)
Ignore All Default Libraries	<edit></edit>
Ignore Specific Default Libraries	
Module Definition File	
Add Module to Assembly	
Embed Managed Resource File	
Force Symbol References	
Delay Loaded Dlls	
Assembly Link Resource	



-adauga in program #include "mpi.h"



-daca dai un run si merge, inseamna ca ai adaugat cu succes libraria

Exemplu cu MPI

```
#include <iostream>
#include "mpi.h"

using namespace std;

MPI_Init(NULL, NULL);
cout << "Orice";

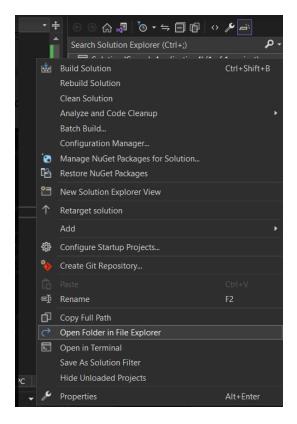
MPI_Finalize();
std::cout << "Hello World!\n";

}</pre>
```

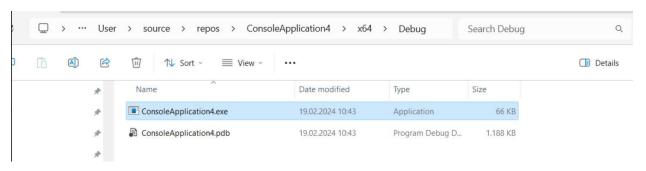
Cum folosim command prompt pentru Debug

-avem codul scris, dam save (nu mai stiu sigur daca e obligatoriu sa salvezi)

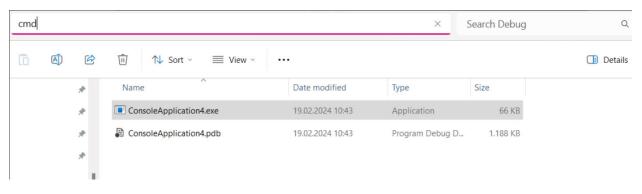
-dupa ce am salvat, apasam click dreapta pe numele proiectului (in cazul de fata e ConsoleApllication4) si selectam "Open Folder in File Explorer"



-mergem la calea x64/Debug, unde selectam cu un singur click .exe-ul



-dupa ce l-am selectat, scriem in bara cmd



-in cmd, avem urmatorul exemplu:

```
C:\Users\User\source\repos\ConsoleApplication4\x64\Debug>mpiexec -n 5 consoleapplication4
Hello World!
Hello World!
Hello World!
Hello World!
Hello World!
C:\Users\User\source\repos\ConsoleApplication4\x64\Debug>
```