

Настройка MPI

Файзрахманов Инсаф Анасович

1. Скачиваем MPI с официального сайта Microsoft: <https://www.microsoft.com/en-us/download/details.aspx?id=57467>

понадобятся оба файла: MSMpiSetup.exe и msmpisdk.msi
устанавливаем их с дефолтными настройками

2. Открываем командную строку от имени администратора и пишем:
set MSMPI

Если всё сможет настроится автоматически, то выйдет что-то такое:

```
C:\Windows\system32>set MSMPI
MSMPI_BIN=C:\Program Files\Microsoft MPI\Bin\
MSMPI_INC=C:\Program Files (x86)\Microsoft SDKs\MPI\Include\
MSMPI_LIB32=C:\Program Files (x86)\Microsoft SDKs\MPI\Lib\x86\
MSMPI_LIB64=C:\Program Files (x86)\Microsoft SDKs\MPI\Lib\x64\
```

Иначе открываем Переменные среды и прописываем их вручную:

Win + R -> systempropertiesadvanced -> Переменные среды

И внизу в "Системные переменные добавляем следующее:

Имя переменной и Значение:

- 1) MSMPI_BENCHMARKS C:\Program Files\Microsoft MPI\Benchmarks\
- 2) MSMPI_BIN C:\Program Files\Microsoft MPI\Bin\
- 3) MSMPI_INC C:\Program Files (x86)\Microsoft SDKs\MPI\Include\
- 4) MSMPI_LIB32 C:\Program Files (x86)\Microsoft SDKs\MPI\Lib\x86\
- 5) MSMPI_LIB64 C:\Program Files (x86)\Microsoft SDKs\MPI\Lib\x64\

Убедитесь что указанные пути существует, иначе вам придется их найти самостоятельно. При этом benchmarks может и не быть, это нормально.

3. Открываем командную строку от имени администратора и пишем: **mpiexec**

Если всё ок, то должно выйти что-то такое:

```
C:\Windows\system32>mpiexec
Microsoft MPI Startup Program [Version 8.1.12438.1084]

Launches an application on multiple hosts.

Usage:

    mpiexec [options] executable [args] [ : [options] exe [args] : ... ]
    mpiexec -configfile <file name>

Common options:
```

Иначе вы что-то сделали не так или этот способ вам не подходит (грусть).

4. Открываем Clion создаём C++ Executable проект
Открываем CmakeLists.txt

Там уже должны быть прописаны некоторые вещи:

cmake_minimum_required(VERSION <номер версии>) - минимальная версия cmake, чтобы этот файл можно было распознать.
project(<название проекта>) - название проекта
set(CMAKE_CXX_STANDARD <номер версии>) - указание по какому стандарту c++ должен компилироваться код

Добавляем строки:

- 1) **find_package(MPI REQUIRED)** - указываем, что надо найти пакет MPI
- 2) **add_executable(MPI1 task1.cpp)** - указываем, что мы должны компилировать код в исполняемый файл (не библиотеку и т.д.). Исполняемый файл должен называться **MPI1**, и нужно скомпилировать код из файла **task1.cpp**.
С таким же успехом для компиляции кода из других файлов вы можете добавить строки:
`add_executable(MPI2, task2.cpp)` или `add_executable(project, main.cpp)`
- 3) **target_link_libraries(MPI1 PUBLIC MPI::MPI_CXX)** - явно указываем какие исполняемые файлы от каких библиотек зависят, в данном случае **MPI1** зависит от **PUBLIC MPI::MPI_CXX**

```
1 cmake_minimum_required(VERSION 3.23)
2 project(PP_MPI)
3
4 set(CMAKE_CXX_STANDARD 14)
5
6 # Add MPI Package to Project
7 find_package(MPI REQUIRED)
8
9 add_executable(project main.cpp)
10 target_link_libraries(project PUBLIC MPI::MPI_CXX)
11
12 add_executable(Task1 task1.cpp)
13 target_link_libraries(Task1 PUBLIC MPI::MPI_CXX)
14
```

5. Вы великолепны, можем открыть исходный код и запустить его прямо из IDE, для проверки работы MPI, можете попробовать запустить следующий код:

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

```
int main(int argc, char *argv[]) {
    int commsize, rank, len;
    char procname[MPI_MAX_PROCESSOR_NAME];
    MPI_Init(&argc, &argv);
    MPI_Comm_size(MPI_COMM_WORLD, &commsize);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Get_processor_name(procname, &len);
    printf("Hello, MPI World! Process %d of %d on node %s.\n", rank, commsize,
        procname);
    MPI_Finalize();
    return 0;
}
```

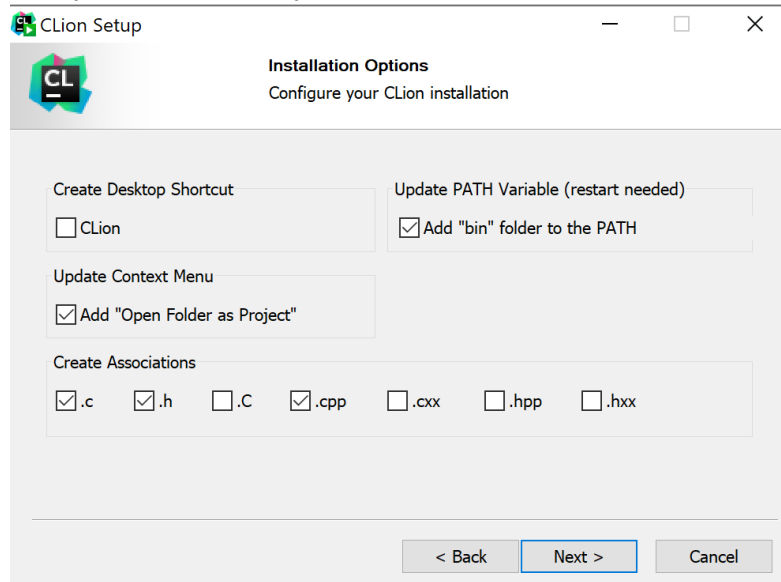
Если Clion еще не установлен:

Настройка Clion и компилятора MinGW

1. Нужно скачать и установить Clion:

<https://www.jetbrains.com/clion/download/#section=windows>

При установке советую поставить флажок на “Add “bin” folder to the PATH”



2. После установки Clion автоматически скачивает и настраивает компилятор Mingw, если этого не случилось, придется это сделать вручную:

Открываем Clion File -> Settings(Ctrl+Alt+S) -> Build,Execution,Deployment -> Toolchains. Нажимаем на плюсики, выбираем MinGw, нажимаем на download. Нас перекидывает на сайт:

Sources

Tarballs for the mingw-w64 sources are hosted on [SourceForge](#).

The latest version from the 10.x series is **10.0.0**.

The latest version from the 9.x series is **9.0.0**.

The latest version from the 8.x series is **8.0.2**.

The latest version from the 7.x series is **7.0.0**.

The latest version from the 6.x series is **6.0.0**.

The latest version from the 5.x series is **5.0.4**.

The old wiki has instructions for building [native](#) and [cross](#) toolchains.

Выбираем SourceForge, нас перекидывает на другой сайт, скачиваем оттуда последнюю версию mingw “Download Latest Version”, если скачивание никак не начинается, подрубите vpn!

Распаковываем скачанный архив в удобное место и в Clion указываем на него путь в Toolset, все остальное обнаружится автоматически:

Name:	MinGW (1)	Add environment ▾
Toolset:	D:\Insaf\Downloads\mingw-w64-v10.0.0	Download...
	✔ Version: w64 9.0	
CMake:	Bundled	
	✔ Version: 3.23.2	
Build Tool:	Detected: ninja.exe	
C Compiler:	Detected: gcc.exe	
C++ Compiler:	Detected: g++.exe	
Debugger:	Bundled GDB	
	✔ Version: 12.1	