

作业六：写Makefile文件

提示：给自己的C++文件们写一个Makefile文件，用Makefile来编译程序，下图供参考

```
#####  
# Makefile of homework  
#####  
  
OBJS=main.o\  
triangle.o\  
  
CXX=g++  
  
OPTION=-O3  
  
serial: ${OBJS}  
    ${CXX} ${OPTION} -o triangle.exe ${OBJS}  
  
.cpp.o:  
    ${CXX} ${OPTION} -c $< -o $@  
  
clean:  
    rm -f *.o *.exe
```

提示：如果有同学会用Cmake也可以，如下图

```
cmake_minimum_required(VERSION 3.9)  
project(tri.exe)  
  
SET(SRC main.cpp triangle.cpp)  
  
option(USE_MPI "Use MPI for parallelization" OFF)  
  
if (USE_MPI)  
    find_package(MPI REQUIRED)  
    include_directories(${MPI_INCLUDE_PATH})  
  
    add_executable(tri_mpi.exe ${SRC})  
    target_link_libraries(tri_mpi.exe ${MPI_LIBRARIES})  
else()  
    add_executable(${PROJECT_NAME} ${SRC})  
endif()
```

作业六：写一个可以用来计时的类

程序结束后，可以打印出时间和调用次数信息，调用方法：第一次调用类的成员函数（静态函数，static关键字），开始计时。第二次调用，结束此次计时。

```
void Mat_Demo::random()
{
    timer::tick("Mat_Demo", "random");
    for(int ic=0; ic<ncols; ++ic)
    {
        for(int ir=0; ir<nrows; ++ir)
        {
            d[ic*nrows+ir]=Random::in_0_1();
        }
    }
    timer::tick("Mat_Demo", "random");
    return;
}
```

作业六：写一个可以用来计时的类

程序结束后，针对不同的类和函数，可以记录每个函数的调用次数，和总时间
当然，每个函数的计时需要通过上文的方法来记录。最后，程序运行结束后，
可以打印出以下信息，此截图供参考。

类名	函数名	总调用时间	总调用次数	每次总调用时间	总时间百分比
CLASS_NAME-----	NAME-----	TIME (Sec) -----	CALLS-----	AVG-----	PER%-----
	total	0.000999	1	0.001	1e+02%
Input	init	0.000161	1	0.00016	16%
Tests	sum_matrix	0.000389	1	0.00039	39%
Mat_Demo	constructor2	2.1e-05	3	7e-06	2.1%
Mat_Read	constructor	1.4e-05	2	7e-06	1.4%
Mat_Read	read_file	0.00014	2	7e-05	14%

作业六：写一个可以用来计时的类

提示：可以用类的静态成员和函数 (static)