

## 1)安装 gtest

```
unzip gtest-1.6.0.zip
cd gtest-1.6.0
./configure
make
```

确认 lib 目录下生成了 libgtest.a 和 libgtest\_main.a,

这两个库是后续测试程序需要链接的库。

## 2)测试 gtest

不妨设我们需要测试 foo.h 中的一个 max 函数(求两个数中较大的一个)

foo.h 文件内如如下:

```
#ifndef __FOO_H__
#define __FOO_H__
int max(int a, int b)
{
    return a>b?a:b;
}
#endif
```

撰写测试程序 foo\_test.cpp 来对 foo 进行测试:

foo\_test.cpp 文件内如如下:

```
#include "gtest/gtest.h"
#include "foo.h"
TEST(foo, max)
{
    EXPECT_EQ(2, max(2,-1));
    EXPECT_EQ(3, max(2,3));
}
int main(int argc, char** argv)
{
    ::testing::InitGoogleTest(&argc, argv);
    return RUN_ALL_TESTS();
}
```

编译 foo\_test.cpp,编译命令如下

```
:g++ -g foo_test.cpp -o foo_test \
-l./gtest-1.6.0/include \
-L./gtest-1.6.0/lib/.libs -Wl,-rpath=./gtest-1.6.0/lib/.libs\
```

`-lgtest -lgtest_main -lpthread`

需要注意的是,要链接 `pthread` 这个库和静态链接库和动态链接库的路径要改。路径为

`libgtest.a` 和 `libgtest.so.0` 的路径

执行结果如下:

```
[=====] Running 1 test from 1 test case.  
[=====] Global test environment set-up.  
[=====] 1 test from foo  
[ RUN] foo.max  
[OK ] foo.max (0 ms)  
[=====] 1 test from foo (0 ms total)  
[=====] Global test environment tear-down  
[=====] 1 test from 1 test case ran. (0 ms total)  
[ PASSED ] 1 test.
```