

gtest 入门

一、用前准备

首先要下载 gtest 代码，google 上不去，在 CSDN 上找了个 1.7.0 版本的，下载解压之后，

```
$ cd gtest/make
$ make
$ ar -rv libgtest.a gtest-all.o
```

得到了 libgtest.a 静态库。

编译的时候要结合着 gtest/include 来用，g++ -Igtest/include -lgtest

二、学习

<http://www.cnblogs.com/coderzh/archive/2009/04/06/1426755.html>

1. 断言：

ASSERT_* 断言失败会退出函数；EXPECT_* 断言失败继续往下执行。

更详细的：

布尔值检查

```
ASSERT_TRUE(condition);    EXPECT_TRUE(condition);    condition == true
ASSERT_FALSE(condition);   EXPECT_FALSE(condition);   condition == false
```

数值型数据检查

```
ASSERT_EQ(val1, val2);     EXPECT_EQ(expected, actual);    val1 == val2
ASSERT_NE(val1, val2);     EXPECT_NE(val1, val2);         val1 != val2
ASSERT_LT(val1, val2);     EXPECT_LT(val1, val2);         val1 < val2
ASSERT_LE(val1, val2);     EXPECT_LE(val1, val2);         val1 <= val2
ASSERT_GT(val1, val2);     EXPECT_GT(val1, val2);         val1 > val2
ASSERT_GE(val1, val2);     EXPECT_GE(val1, val2);         val1 >= val2
```

浮点型检查

```
ASSERT_FLOAT_EQ(expected, actual);    EXPECT_FLOAT_EQ(expected, actual);
ASSERT_DOUBLE_EQ(expected, actual);    EXPECT_DOUBLE_EQ(expected, actual);
```

字符串检查

```
ASSERT_STREQ(expected_str, actual_str);
EXPECT_STREQ(expected_str, actual_str);
ASSERT_STRNE(str1, str2);    EXPECT_STRNE(str1, str2);
ASSERT_STRCASEEQ(expected_str, actual_str);
EXPECT_STRCASEEQ(expected_str, actual_str);
ASSERT_STRCASENE(str1, str2);    EXPECT_STRCASENE(str1, str2);
```

预测断言

```
ASSERT_PRED1(pred1, val1);    EXPECT_PRED1(pred1, val1);    pred1(val1) == true
ASSERT_PRED2(pred2, val1, val2);    pred2(val1, val2) == true
EXPECT_PRED2(pred2, val1, val2);    pred2(val1, val2) == true
```

参数数量最多到 5 个，即 PRED5。

2. 测试覆盖范围

语句覆盖：语句覆盖就是设计若干个测试用例，运行被测试程序，使得每一条可执行语句至少执行一次。

分支覆盖：设计若干个测试用例，运行所测程序，使程序中每个判断的取真分支和取假分支至少执行一次。

条件覆盖：设计足够的测试用例，运行所测程序，使程序中每个判断的每个条件的每个可能取值至少执行一次。

3. 应用，具体情况具体分析。