

# Documentação dos testes

August 29, 2016

No projeto há uma classe genérica de testes (TestGeneric) que executa a solução desenvolvida com todos os arquivos de testes que estão disponíveis no site oficial da maratona. Cada solução é comparada com as soluções dos arquivos de testes. Se alguma solução estiver errada será informado, pela saída padrão, o nome do arquivo de teste em que a aplicação falhou, e o teste de unidade irá falhar.

Para usar esta classe de teste é preciso setar algumas configurações, setadas pela classe Configuration. Estas configurações são: o ano da maratona, a fase ('final' ou 'fase1'), a etapa ('aquecimento' ou 'competicao'), e a letra maiúscula que indentifica o problema.

## 1 Requisitos

As strings de configuração identificarão o pacote da classe que deverá ser testada, também identificarão o diretório onde estão os arquivos de testes.

Portanto, para que o teste funcione, os seguintes requisitos devem ser cumpridos:

- O pacote da classe deve ter o seguinte nome:  
org.ufla.maratonadeprogramacao.\_<year>.<phase>.<step>
- O diretório dos arquivos de testes, dentro do projeto, deve ter o seguinte nome:  
problemas/\_<year>/<phase>/<step>/<problem>

Onde:

- <year>: o ano da maratona;
  - <phase>: a fase, pode ser 'fase1' ou '\_final';
  - <step>: a etapa, pode ser 'aquecimento' ou 'competicao';
  - <problem>: letra maiúscula que identifica o problema.
- O nome da classe deve ser a letra maiúscula que identifica o problema.
  - Os arquivos de testes da maratona são identificados como entrada ou saída pela extensão '.in' ou '.sol' respectivamente.

```

3 import java.io.IOException;
4
5 import org.junit.Test;
6 import org.ufla.maratonadeprogramacao.generictest.Configuration;
7 import org.ufla.maratonadeprogramacao.generictest.GenericTest;
8
9 public class AquecimentoTest {
10
11     String year = "2015";
12     String phase = "fase1";
13     String step = "aquecimento";
14
15     @Test
16     public void testA() throws IOException, InterruptedException {
17         GenericTest testGeneric = new GenericTest(new Configuration(year,
18             phase, step, "A"));
19         testGeneric.test();
20     }
21
22     @Test
23     public void testB() throws IOException, InterruptedException {
24         GenericTest testGeneric = new GenericTest(new Configuration(year,
25             phase, step, "B"));
26         testGeneric.test();
27     }
28

```

Figure 1: Exemplo de teste de unidade de um problema.

## 2 Exemplo

Na Figura 1 é apresentado um exemplo de uso da classe de teste. Pode-se perceber que no método `testA()` é criado um `GenericTest`, passando uma instância da classe `Configuration` com as seguintes configurações: `year='2015'`, `phase='fase1'`, `step='aquecimento'`, `problem='A'`. Portanto quando executar o método `test()` do `GenericTest` instanciado ele irá executar a classe **`org.ufla.maratonadeprogramacao._2015.fase1.aquecimento.A`** com todos os arquivos de entrada, `.in`, disponíveis no diretório `/problemas/_2015/fase1/aquecimento/A/` e irá comparar a saída com os respectivos arquivos de saída, `.sol`.