



Introdução à Programação Orientada a Objetos (TSI32A)

Professor: Ivan Luiz Salvadori

Trabalho 2

Atenção: Esta atividade será avaliada e corresponderá nota de acordo com o plano de ensino da disciplina.

1. Escreva um programa em Java capaz de simular competições entre atletas de três modalidades distintas, sendo elas:

- Ginástica
- Nataç  o
- Halterofilismo

Os atletas possuem diferentes atributos que definem o seu desempenho na competi  o. Os atributos considerados s  o:

- peso
- flexibilidade
- resist  ncia
- for  a
- f  lego

Al  m dos atributos, atletas podem possuir les  es decorrentes de seus treinamentos e competi  es anteriores. A quantidade de les  es interfere negativamente no desempenho dos atletas.

O desempenho dos atletas s  o simulados por f  rmulas que consideram os atributos dos atletas em cada modalidade. O c  lculo para cada modalidade deve ser realizado da seguinte forma:

Desempenho dos ginastas:

$$d = 1 + (\text{for  a} * 0.6) + (\text{resist  ncia} * 0.6) + (\text{f  lego} * 0.5) + (\text{flexibilidade} * 1.0) - (\text{peso}/100) - (\text{nrLes  es}/10)$$

Desempenho dos nadadores:

$$d = 1 + (\text{for  a} * 0.5) + (\text{resist  ncia} * 0.6) + (\text{f  lego} * 1.0) + (\text{flexibilidade} * 0.3) - (\text{peso}/100) - (\text{nrLes  es}/10)$$

Desempenho dos halterofilistas:

$$d = 1 + (\text{for  a} * 1.0) + (\text{resist  ncia} * 0.6) + (\text{f  lego} * 0.4) + (\text{flexibilidade} * 0.3) + (\text{peso}/100) - (\text{nrLes  es}/10)$$

As listas dos atletas estão disponíveis por meio de três arquivos JSON distintos. Cada arquivo contém os atletas que irão participar de cada uma das modalidades esportivas. O exemplo a seguir ilustra o conteúdo do arquivo:

```
1 [
2   {
3     "nome": "Matilda Maydwell",
4     "dataNascimento": "31/10/1982",
5     "peso": 0.3,
6     "flexibilidade": 0.08,
7     "resistencia": 0.5,
8     "folego": 0.58,
9     "forca": 0.96,
10    "numeroLesoes": 7
11  },
12  {
13    "nome": "Ronni Eadington",
14    "dataNascimento": "22/09/2003",
15    "peso": 0.54,
16    "flexibilidade": 0.99,
17    "resistencia": 0.81,
18    "folego": 0.2,
19    "forca": 0.26,
20    "numeroLesoes": 6
21  }
22 ]
```

Os atletas contidos nos arquivos devem ser inscritos nas suas respectivas competições, sendo necessário cumprir os seguintes requisitos:

- Atletas entre 18 e 45 anos de idade (calculado no momento da simulação)
- Não possuírem mais de 10 lesões

Os atletas que cumprem os requisitos são incluídos na competição. Entretanto, os atletas não admitidos deverão ser excluídos da competição. Ao término da simulação, o programa deve gerar um arquivo JSON para informar o resultado de cada modalidade. O nome do arquivo deve obedecer o seguinte padrão: *resultado-<modalidade>.json*. Os atletas que não cumprem os requisitos para participar das competições também devem ser registrados em um arquivo JSON, que deve ser nominado pelo padrão: *inscricoes-negadas-<modalidade>.json*. Os exemplos a seguir ilustram os arquivos de resultado esperados para competição de ginástica:

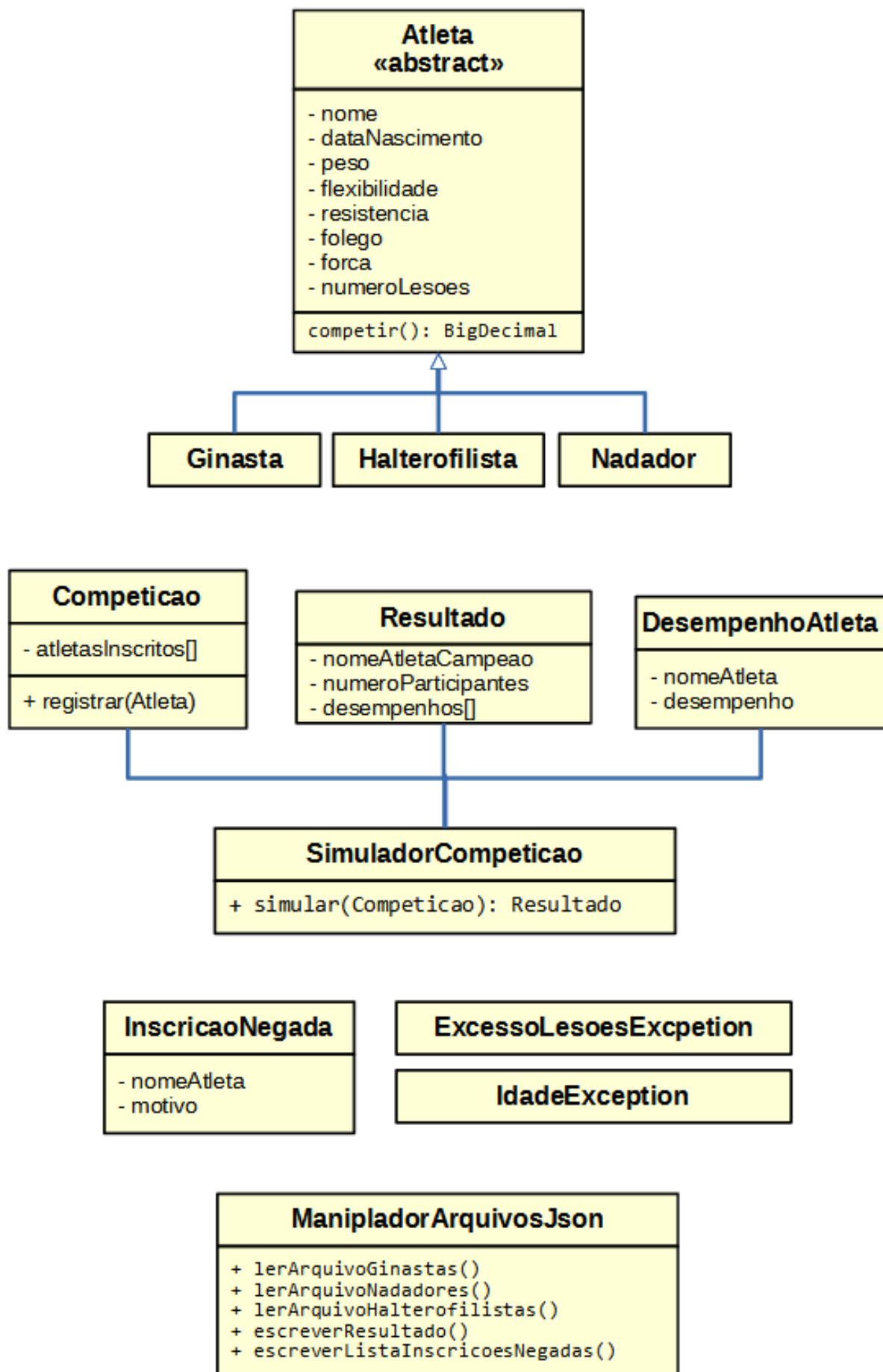
```
1 {
2   "nomeAtletaCampeao": "Kippie Buckthorp",
3   "numeroParticipantes": 3,
4   "desempenhos": [
5     {
6       "nomeAtleta": "Matilda Maydwell",
7       "desempenho": 1.543
8     },
9     {
10      "nomeAtleta": "Ronni Eadington",
11      "desempenho": 2.1266
12    },
13    {
14      "nomeAtleta": "Kippie Buckthorp",
15      "desempenho": 2.2949
16    }
17  ]
18 }
```

resultado-ginastica.json

```
1 {
2   "numeroInscricoesNegadas": 3,
3   "inscricoesNegadas": [
4     {
5       "nomeAtleta": "Niccolo Bredee",
6       "motivo": "Atleta menor de idade"
7     },
8     {
9       "nomeAtleta": "Benni Huey",
10      "motivo": "Idade do atleta é superior ao permitido"
11    },
12    {
13      "nomeAtleta": "Daron Blacker",
14      "motivo": "Atleta com excesso de lesoes"
15    }
16  ]
17 }
```

inscricoes-negadas-ginastica.json

Diagrama sugerido:



Instruções gerais:

- O programa deve processar os arquivos JSON disponibilizados para a atividade.
- O caminho dos arquivos JSON deve ser o mesmo do **arquivo JAR executável**.
- O programa deve seguir os exemplos da versão do professor.
- Esta atividade pode ser desenvolvida em grupo de até 3 alunos.
- A entrega deve ser feita no Moodle da disciplina através da submissão dos seguintes arquivos:
 - Arquivo de texto com os nomes, e-mails e RA dos autores.
 - Código-fonte do projeto (pasta compactada ZIP)
 - Arquivo JAR executável
- A entrega deve ser feita por apenas um membro do grupo.
- A entrega deve ser realizada através do Moodle até o dia **28/11/2022** 23h59.

Toledo – PR, 11/11/2022