

## Missão 1: Prog3, o Retorno

Jogadores, organizados em seus times e munidos de suas novas armas (Linguagens de Programação modernas), sua missão no jogo de LP é:

1. Repetir a missão já cumprida em Prog3 (simplificada) com sua nova arma;
2. Realizar uma análise técnica de sua nova arma;
3. Apresentar o cumprimento da missão e a análise técnica aos demais times e entregá-la ao Oráculo.

### 1. Repetindo a missão de Prog3

Após consulta aos diferentes times que participam do jogo, as seguintes missões de Prog3 foram analisadas:

Missão	Período	Professor(a)	Link para Especificação
Credenciamento PPGI	2019/2	Jordana	<a href="http://www.inf.ufes.br/~jssalamon/wp-content/uploads/disciplinas/prog3/trabalho/Java/Prog3-Especificação%20do%20Trabalho_Java.pdf">http://www.inf.ufes.br/~jssalamon/wp-content/uploads/disciplinas/prog3/trabalho/Java/Prog3-Especificação%20do%20Trabalho_Java.pdf</a>
Inventário de Mídias	2019/2	Jordana	<a href="http://www.inf.ufes.br/~jssalamon/wp-content/uploads/disciplinas/prog3/trabalho/CPP/Trabalho_PROG3.pdf">http://www.inf.ufes.br/~jssalamon/wp-content/uploads/disciplinas/prog3/trabalho/CPP/Trabalho_PROG3.pdf</a>
Sistema Eleitoral	2018/2	João Paulo	<a href="https://www.dropbox.com/s/ktpvzo6uy2w9hs2/prog-iii-2018-02-trabalho.pdf?dl=0">https://www.dropbox.com/s/ktpvzo6uy2w9hs2/prog-iii-2018-02-trabalho.pdf?dl=0</a>

No jogo de LP, as missões acima são simplificadas da seguinte forma:

- Credenciamento PPGI / Inventário de Mídias:
  - Não é necessário produzir todos os relatórios, apenas 2 deles, à escolha;
  - Não é necessário implementar serialização e desserialização;
  - Não é necessário tratar erros/exceções/inconsistências (considere entradas sempre válidas).
- Sistema Eleitoral:
  - Não é necessário produzir todos os 8 relatórios, apenas os relatórios "Número de vagas", "Total de votos nominais" e mais 2 outros, à escolha;
  - Não é necessário produzir o relatório que descreve sua implementação;
  - Não é necessário tratar erros/exceções/inconsistências (considere entradas sempre válidas).

Além das simplificações acima, não será feita correção automatizada dos códigos produzidos, de modo que as restrições para a correção automatizada (ex.: que os relatórios sejam formatados exatamente como esperado pelo script de testes ou pelo BOCA) não se aplicam aqui. Também não há aqui nenhuma condição de ponto extra. A forma de comprovação por parte dos times de que cumpriram a missão é detalhada no item 3 desta especificação.

## 2. Análise técnica de sua arma

Após experimentarem a nova arma no cumprimento da missão de Prog3, o time deve realizar uma análise técnica (teórica) da arma utilizada, respondendo às seguintes perguntas:

1. Como instalar e começar a usar?
2. Quais são os processos de tradução utilizados?
3. Em que paradigmas se encaixa?
4. Os nomes são sensíveis à capitalização?
5. Quais os caracteres aceitos em um nome?
6. Existe alguma restrição de tamanho para nomes?
7. Como é a questão das palavras-chave x palavras reservadas?
8. É possível definir uma variável anônima? Mostre exemplo.
9. A vinculação de tipos (tipagem) é estática ou dinâmica?
10. Quais categorias de variável (Sebesta, Seção 5.4.3) apresenta? Mostre exemplos.
11. Permite ocultamento de nomes (variáveis) em blocos aninhados? Mostre exemplo.
12. Permite definir constantes? Vinculação estática ou dinâmica? Mostre exemplos.
13. Quais os tipos oferecidos? Mostre exemplos de definição de variáveis de cada tipo.
14. Existe o tipo função? São cidadãos de primeira classe? Mostre exemplo.
15. Possui ponteiros ou referências? Permite aritmética de ponteiros?
16. Oferece coletor de lixo? Se sim, qual a técnica utilizada?
17. É possível quebrar seu sistema de tipos (forçar erro de tipo)? Mostre exemplo.
18. Quais os operadores oferecidos? Mostre exemplo de uso de cada operador.
19. Permite sobrecarga de operadores? Mostre exemplo.
20. Quais operadores funcionam com avaliação em curto-circuito?
21. O operador de atribuição funciona como uma expressão?
22. Quais as estruturas de controle (seleção, iteração) oferecidas? Mostre exemplos.
23. Quais sentenças de desvio incondicional oferecidas? Mostre exemplos.
24. Quais os métodos de passagem de parâmetros oferecidos? Mostre exemplos.
25. Permite sobrecarga de subprogramas? Mostre exemplo.
26. Permite subprogramas genéricos? Mostre exemplo.
27. Como é o suporte para definição de Tipos Abstratos de Dados? Mostre exemplo.
28. Permite TADs genéricos/parametrizáveis? Mostre exemplo.
29. Quais as construções de encapsulamento oferecidas? Mostre exemplos.
30. Quais tipos de polimorfismo suporta? Mostre exemplos.
31. Permite herança de tipos? Herança múltipla? Mostre exemplo.
32. Permite sobrescrita de subprogramas? Mostre exemplo.
33. Permite a definição de subprogramas abstratos? Mostre exemplo.
34. Oferece mecanismo de controle de exceções? Mostre exemplo.
35. Possui hierarquia de exceções controlada, como em Java? Qual a raiz?
36. Categoriza as exceções em checadas e não-checadas? Como?
37. Obriga a declaração de exceções lançadas para fora de um subprograma?
38. Como você avalia a LP usando os critérios do Sebesta (Seção 1.3)?
39. Como você avalia a LP usando os critérios do Varejão?

Dada a forma de apresentação da análise acima, detalhada a seguir, sugerimos o uso de slides como formato de resposta.

### 3. Apresentando a missão cumprida e a análise da arma

Nas datas definidas no calendário, os times terão **25 minutos** para mostrar aos demais times:

- Como a missão foi cumprida com o uso da arma escolhida;
- Os aspectos mais interessantes da análise técnica da arma.

Visto que 25 minutos não são suficientes para apresentar as respostas a todas as perguntas listadas acima (e os exemplos associados), o time deve escolher os aspectos mais interessantes da análise a serem demonstrados aos colegas.

O dia da última apresentação de missão no calendário fica estabelecido como prazo para entrega do código-fonte e da análise da arma ao Oráculo. Um membro de cada time deve dirigir-se ao [Templo de AVA](#) e entregar os artefatos até a data limite.

### 4. Recompensas

Os membros dos times serão recompensados da seguinte forma, totalizando um máximo de 100 Turings:

- Até 36 Turings pelo código-fonte (análise principalmente objetiva);
- Até 39 Turings pela análise da arma (até 1 Turing por questão);
- Até 25 Turings pela apresentação aos colegas.

### 5. Observações finais

Caso haja algum erro neste documento, serão publicadas novas versões e divulgadas erratas durante uma partida do jogo. É responsabilidade do aluno manter-se informado, frequentando as partidas ou acompanhando as novidades divulgadas no Templo de AVA.