



Instituto Superior de Engenharia de Coimbra

Licenciatura em Engenharia Informática

Programação Orientada a Objetos

Leandro Adão Fidalgo | a2017017144

Pedro dos Santos Alves | a2019112789

Laboratório P6

Trabalho Prático

Coimbra, 8 de dezembro de 2020

Índice

Introdução.....	3
Organização do código apresentado.....	4
1. Quais foram as classes consideradas na primeira versão da aplicação que foi testada?	4
2. Quais os conceitos/classe que identificou ao ler o enunciado?	4
3. Relativamente a duas das principais classes da aplicação, identifique em que classes ou partes do programa são criados, armazenados e destruídos os seus objetos.	5
4. Indique um exemplo de uma responsabilidade atribuída a uma classe que esteja de acordo com a orientação dada acerca de Encapsulamento.	5
5. De entre as classes que fez, escolha duas e justifique por que considera que são classes com objetivo focado, coeso e sem dispersão.	5
6. Relativamente à aplicação entregue, quais as classes que considera com responsabilidades de interface com o utilizador e quais as que representam a lógica?.....	5
7. Identifique o primeiro objeto para além da camada de interação com o utilizador que recebe e coordena uma funcionalidade de natureza lógica?	5
8. A classe que representa a envolvente de toda a lógica executa em pormenor muitas funcionalidades, ou delega noutras classes? Indique um exemplo em que esta classe delega uma funcionalidade noutra classe.	6
9. Dê um exemplo de uma funcionalidade que varia conforme o tipo do objeto que a invoca. Indique em que classes e métodos está implementada esta funcionalidade. (Não é necessário responder a esta pergunta na meta 1).	6
10. Apresente as principais classes da aplicação através da seguinte informação:	6
Funcionalidades implementadas	7

Introdução

O presente relatório descreve o projeto desenvolvido pelos alunos: Leandro Fidalgo e Pedro Alves, no âmbito da disciplina de Programação Orientada a Objetos da Licenciatura em Engenharia Informática do Instituto Superior de Engenharia de Coimbra.

O tema do projeto é o desenvolvimento de um jogo do tipo single-player sobre conquista e expansão territorial, sendo este desenvolvido em C++.

O objetivo do presente trabalho é consolidar todos os conhecimentos adquiridos nas aulas teóricas e práticas ao longo de todo o semestre.

Organização do código apresentado

1. Quais foram as classes consideradas na primeira versão da aplicação que foi testada?

A primeira versão tinha as seguintes classes e principais funcionalidades:

- Jogo
 - Indica o estado do jogo;
 - Trata dos comandos provenientes do utilizador;
 - Inicia o jogo;
 - Trata do decorrer do jogo (fases).
- Mundo
 - Guarda e trata da informação de todos os territórios;
 - Guarda a informação do império do jogador.
- Imperio_Jogador
 - Guarda os territórios conquistados;
 - Contém a informação do império do jogador.
- Territorio
 - Contém a informação básica do território.
- Territorio_Inicial
 - Indica o território inicial do jogador.
- Continente
 - Contém a informação básica do continente.
- Castelo
- Duna
- Fortaleza
- Mina
- Montanha
- Planície
- Ilha
 - Contém informação básica da Ilha.
- Pescaria
- Refugio_Piratas
- Tecnologias
 - Contém a informação sobre as diversas tecnologias e o custo das mesmas.

2. Quais os conceitos/classe que identificou ao ler o enunciado?

Os conceitos/classe identificados foram:

- Imperio do jogador
- Mundo
- Território
- Território inicial
- Continente
 - Planície
 - Montanha

- Fortaleza
 - Mina
 - Duna
 - Castelo
- Ilha
 - Refúgio dos Piratas
 - Pescaria
- Tecnologias
- Jogo

3. Relativamente a duas das principais classes da aplicação, identifique em que classes ou partes do programa são criados, armazenados e destruídos os seus objetos.

Territorio: os objetos desta classe são criados na classe Jogo, armazenados e destruídos na classe Mundo.

Imperio_Jogador: os objetos desta classe são criados, armazenados e destruídos na classe Mundo.

4. Indique um exemplo de uma responsabilidade atribuída a uma classe que esteja de acordo com a orientação dada acerca de Encapsulamento.

A responsabilidade “adicionar territórios” está atribuída à classe Mundo porque tem a coleção de Territórios.

5. De entre as classes que fez, escolha duas e justifique por que considera que são classes com objetivo focado, coeso e sem dispersão.

Classe Mundo: tem dados e responsabilidades relativos apenas ao Império e Territórios, como a adição de territórios, conquista e a sua listagem.

Classe Jogo: tem dados e responsabilidades relativos ao Mundo, tal como a manipulação deste através dos diversos comandos.

6. Relativamente à aplicação entregue, quais as classes que considera com responsabilidades de interface com o utilizador e quais as que representam a lógica?

Responsabilidade de interface: Jogo

Responsabilidades da lógica da aplicação: Restantes classes já descritas acima.

7. Identifique o primeiro objeto para além da camada de interação com o utilizador que recebe e coordena uma funcionalidade de natureza lógica?

As ordens vindas da camada de interação com o utilizador são recebidas e processadas por um objeto da classe Mundo.

8. A classe que representa a envolvente de toda a lógica executa em pormenor muitas funcionalidades, ou delega noutras classes? Indique um exemplo em que esta classe delega uma funcionalidade noutra classe.

A classe Mundo representa a envolvente de toda a lógica. Para adicionar um território ainda não conquistado ao Império, delega a adição desse território ao Império.

9. Dê um exemplo de uma funcionalidade que varia conforme o tipo do objeto que a invoca. Indique em que classes e métodos está implementada esta funcionalidade. (Não é necessário responder a esta pergunta na meta 1).

10. Apresente as principais classes da aplicação através da seguinte informação:

Classe: Jogo

Responsabilidades:

- Ler comandos do utilizador;
- Estado do jogo (turno, ano, fator sorte, indicação se o jogo está a correr);
- Trata do decorrer do jogo (fases).

Colaborações: Mundo

Classe: Mundo

Responsabilidades:

- Guardar os territórios;
- Adicionar territórios.

Colaborações: Jogo, Império, Território

Classe: Império

Responsabilidades:

- Contém a informação do Império (Tecnologias, armazém, cofre, força militar, fator sorte);
- Trata dos territórios conquistados.

Colaborações: Mundo, Território, Tecnologia

Classe: Território

Responsabilidades:

- Contém a informação dos territórios (nome, pontos de vitória, resistência, criação de produtos, criação de ouro).

Colaborações: Mundo, Império

Classe: Tecnologia

Responsabilidades:

- Contém a informação sobre as tecnologias (custos e se as tecnologias foram adquiridas ou não).
- Trata da compra das tecnologias.

Colaborações: Império

Funcionalidades implementadas

Componente do trabalho	Realizado	Realizado parcialmente	Não realizado
Configuração do mundo através de comandos (cria)	x		
Configuração do mundo através de comandos (carrega)	x		
Conquista de territórios (comando conquista)	x		
Visualização dos dados do jogo (comando lista)	x		