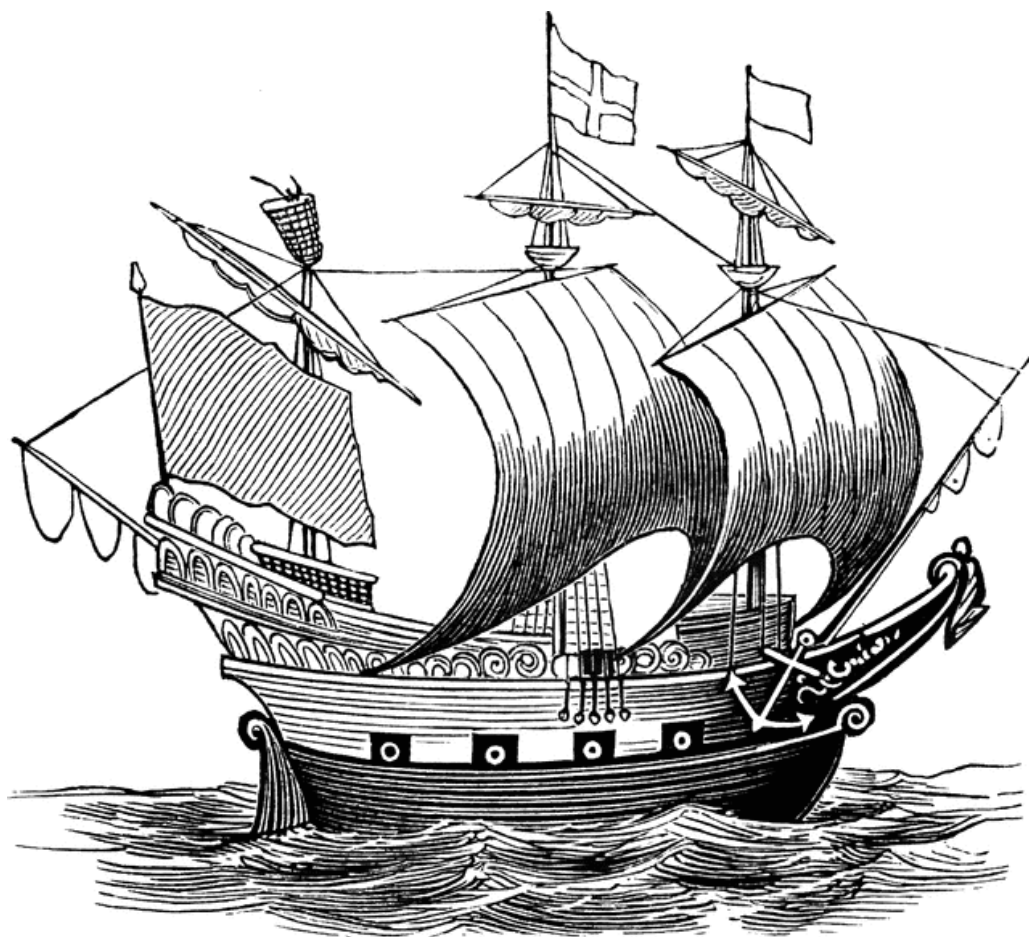




Departamento de Engenharia Informática e de Sistemas
Licenciatura em Engenharia Informática – Ramo de Desenvolvimento de Aplicações

Programação Orientada a Objetos
2018/2019



Trabalho prático: Navy Expeditions
1ª Meta

Trabalho Realizado por:
Diogo Pereira de Lima nº21240010
Nuno Vieira Gonçalves nº21240426

Conteúdo

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

Introdução

No âmbito da cadeira de Programação Orientada a Objetos foi proposto aos alunos a realização de um jogo sobre expedições marítimas, com pescadores, piratas e batalhas navais.

Nesta primeira meta foi requisitada a configuração do mapa, a interpretação de comandos através de um ficheiro de textos, a opção de compra de navios e o movimento aleatório dos navios.

Organização do código apresentado

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

- Configuration
 - Tudo o que seja de configuração do jogo, como o mapa e os valores iniciais.
- Game
 - Tratamento de comandos.
- Jogador
 - Onde estão armazenados os dados de jogo do jogador
- Navios
 - Onde está armazenada toda a informação sobre os navios, como o seu id, número de soldados, quantidade de peixe e mercadoria e a água existente
- Escuna, Fragata, Galeao, Veleiro e late
 - Tipos de navios existentes
- Porto
- Posição
 - Onde vai ser guardada a posição de cada objeto no mapa
 - e

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

- Interface
- Jogador
- Navios
- Portos
- Veleiro, Fragata, Galeão e Escuna
- Posição

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.

As duas principais classes da aplicação são a Configuration e o Game.

Jogador: os objetos desta classe são criados, armazenados e destruídos na classe Game

Navios: os objetos desta classe são criados, armazenados e destruídos na classe Game

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

A responsabilidade dos navios é da classe Jogador pois tem o vetor de navios existentes.

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 Navios: tem os dados relativos a todos os navios, e é simples e objetivo

Classe Configuration: está focada na inicialização do jogo e dos parâmetros iniciais

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

Classe Configuration está responsável pela lógica de configuração.

Classe Game está responsável pela lógica do jogo e tem a interação com o utilizador através do menu

7 – Identifique o primeiro objecto para além da camada da 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 Configuration.

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.

Todas as classes lidam com lógica, a classe Game lida com a maior parte da lógica, mas com o auxílio de outras funções, uma delas sendo a classe Jogador, que é também uma classe com lógica. Por exemplo, é na classe Jogador que está a função moverNavios();

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

Vazio.

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

Classe: Configuration

Responsabilidades: Inicialização do jogo e dos parâmetros iniciais

Colaborações: Vazio

Classe: Game

Responsabilidades: Simulação do jogo e visualização do mapa

Colaborações: Configuration, Porto, Jogador

Classe: Jogador

Responsabilidade: Armazenar informação do jogo relativa ao jogador e movimentação dos navios

Colaborações: Navio, Veleiro, Fragata, Galeao, Escuna, late

Classe: Navio

Responsabilidade: Armazenar informação sobre os navios

Colaborações: Posicao

Classe: Posicao

Responsabilidade: Guardar a posição dos navios

Colaborações: Vazio

Funcionalidades implementadas

| Componente do trabalho | Realizado | Realizado Parcialmente | Não Realizado |
|-------------------------------------|-----------|------------------------|---------------|
| Configuração/leitura do mapa | X | | |
| Visualização do mapa | X | | |
| Interpretação de comandos | X | | |
| Compra de navios | X | | |
| Visualização de movimento de navios | X | | |

Conclusão

Com a conclusão desta meta demos o primeiro passo na realização do trabalho, o que nos fez olhar para o trabalho que realizamos até agora. tendo em conta que completamos todos os requisitos necessários para a entrega e não encontramos nenhum erro no código, assumimos que vamos num bom caminho e preparados para a 2ª e última meta.