



Instituto Federal de Minas Gerais - Campus Ouro Branco

Curso: Téc. Informática

Disciplina: Programação II

Professor: Saulo Henrique Cabral Silva (saulo.cabral@ifmg.edu.br)

Trabalho Prático 5

Batalha Naval

Valor: 13 pontos

Data da entrega: 22/09/2023

Objetivos: Consiste em rever conceitos básicos de programação bem como a prática de programação em Java, utilizando matrizes, POO, tratamento de eventos, laços de repetição e interfaces gráficas.

Descrição:

Seu José é um apaixonado pelo jogo de Batalha Naval. Durante os últimos anos, Seu José e seus amigos se reúnem nos finais de semana para colocar o papo em dia, e claro, jogar Batalha Naval. O jogo de Batalha Naval é um jogo tradicional no estilo papel e caneta, jogado por dois jogadores. Cada jogador possui uma matriz retangular, onde são representadas as suas embarcações. Cada Jogador distribui em sua matriz as suas embarcações da forma como desejar. Após posicionar todas as embarcações, o objetivo do jogo é adivinhar onde o adversário distribuiu as suas embarcações e afundá-las utilizando "tiros" (tentativas). Uma embarcação é afundada quando todas as células que a formam forem atingidas/descobertas pelo adversário. Ganha o jogo quem afundar primeiro todas as embarcações do adversário.

Bom, durante todos estes anos jogando batalha naval, Seu José decidiu abandonar os tabuleiros, papeis e as canetas utilizadas durante o jogo, e pede a você caro estudante para desenvolver uma versão digital desse jogo. Veja abaixo as especificações que Seu José espera no Jogo de Batalha Naval.

Mínimo:

- A versão mínima que deve ser implementada no trabalho deve possuir um tabuleiro de no mínimo 5 x 10;
- As embarcações disponíveis para distribuir no tabuleiro são: 1 cruzador (duas células), 1 fragata (três células) e porta-aviões (quatro células);
 - Trabalhos que adicionem mais embarcações receberam pontos extra.
- Seu programa deve permitir que os jogadores informem inicialmente a posição de distribuição das embarcações no tabuleiro. Seu programa deve ser construído de forma que aceite somente posições válidas para alocação das embarcações;
 - Trabalhos que não verifiquem isso PERDERAM muitos pontos!

- Nessa versão do jogo, será uma disputa entre homem x máquina (cpu/IA).
 - O algoritmo deve permitir que o usuário(humano) informe a posição de um possível disparo.
 - O o programa deve permitir que a máquina (cpu), gere (“chute”) um disparo em uma posição aleatória.
- A cada disparo efetuado por um dos jogadores (*humano* ou *máquina*), deve ser informado se o tiro atingiu alguma arma do adversário.
- No decurso da partida, o seu programa deverá verificar e informar quando houver um vencedor. Nenhum tiro poderá ser disparado após este cenário.
- Crie uma representação visual do jogo que seja intuitiva para que os usuários possam visualizar seus ataques (GUI).

Para saber e conhecer maiores detalhes sobre a versão tradicional do jogo de Batalha Naval, visite: [https://pt.wikipedia.org/wiki/Batalha_naval_\(jogo\)](https://pt.wikipedia.org/wiki/Batalha_naval_(jogo)).

1 2 3 4 5 6 7 8 9 10

A

B

C

D

E

Posição inicial:

Porta-aviões (4x): ☒ H ☐ V

Fragata (3x): ☐ H ☒ V

Cruzador (2x): ☒ H ☐ V

Iniciar

1 2 3 4 5 6 7 8 9 10

A

B

C

D

E

Disparo em:

Atirar!!!

Acertos Player: 2 Acertos IA: 0

Figura 1: esboço da solução a ser desenvolvida por vocês

O que deve ser entregue:

1. Código fonte do programa em Portugol Studio (bem *identada* e comentada).
2. Documentação do trabalho. Entre outras coisas, a documentação deve conter:
 - 2.1. Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do programa.
 - 2.2. Implementação: descrição sobre a implementação do programa. Deve ser detalhada a estrutura de dados utilizada (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, o formato de entrada e saída de dados, bem como decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado.
 - 2.3. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
 - 2.4. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso
3. Formato: mandatoriamente em PDF (<http://www.pdf995.com/>).

Obs1: Apesar desse trabalho ser bem simples, a documentação pedida segue o formato da documentação que deverá ser entregue nos próximos trabalhos.

Obs2: Consulte as dicas do Prof. Nívio Ziviani de como deve ser feita uma boa implementação e documentação de um trabalho prático: [DOCUMENTO](#)

Como deve ser feita a entrega:

A entrega **DEVE** ser feita por email na forma de um único arquivo *zipado*, contendo o código, os arquivos, o executável e a documentação.

Comentários Gerais:

- Comece a fazer este trabalho logo, enquanto o problema está fresco na memória e o prazo para terminá-lo está tão longe quanto jamais poderá estar;
- Clareza, *indentação* e comentários no programa também vão valer pontos;
- O trabalho pode ser realizado em grupo (**Máximo 3 alunos**);
- Trabalhos copiados (e **FONTE**) terão nota ZERO;
- Trabalhos entregues em atraso serão aceitos, todavia a nota atribuída ao trabalho será zero
- Evite discussões inúteis com o professor em tentar postergar a data de entrega do referido trabalho.