

Descrição do Trabalho

Escolher um dos jogos abaixo e implementar em Python 3:

A) JOGO DE BATALHA NAVAL

- Jogo de batalha naval do jogador contra o computador
- Deve haver dois tabuleiros: um do jogador e outro do computador, ambos de 10 x 10 posições
- Cada um dos participantes possui 5 embarcações. Cada uma ocupa uma única posição no tabuleiro
- No início de cada partida, o jogador e o computador devem determinar as posições de suas embarcações
- Durante a partida deve ser mostrado o tabuleiro do adversário, indicando posições já utilizadas: água ou embarcação afundada
- Cada jogador tem direito a dar um tiro por vez. Caso acerte uma embarcação, tem direito a dar outro tiro
- O vencedor é aquele que conseguir afundar todas as embarcações do adversário
- O programa deve permitir, através de um comando especial, que o jogador veja o tabuleiro do computador (para testes)

B) JOGO DE CAMPO MINADO

- Jogo de campo minado contra o computador
- O jogador indica o tamanho do campo minado (no máximo 10 x 10), bem como o número de minas a serem escondidas
- O computador esconde a quantidade de minas selecionadas aleatoriamente dentro do campo estabelecido e marca todas as posições com o símbolo '*'
- Durante cada tentativa o jogador pode:
 - marcar uma posição como mina (ficando marcada com a letra 'M')
 - abrir uma posição (ficando marcada com a quantidade de minas vizinhas nas 8 posições vizinhas)
- O jogador perde se abrir uma posição que contenha uma mina
- O jogador vence se conseguir marcar com a letra 'M' todas as posições corretas

Observações importantes:

- O trabalho deve ser realizado individualmente por cada aluno.
- O trabalho deverá ser entregue, única e exclusivamente, por meio de um repositório no GitHub.
- Abuse dos comentários para explicar o código.
- É responsabilidade do aluno garantir que o programa compile e execute corretamente, corrigindo eventuais erros.
- Em caso de erro de sintaxe (compilação), o peso final do trabalho pode sofrer uma redução de até 50% do peso inicial.
 - Em caso de redução do peso final, a documentação adicional, como comentários adicionais no programa fonte pode acrescentar o peso final em até 50% do peso reduzido.
- Todos os trabalhos estão sujeitos a apresentação individual caso seja solicitada.
- Em caso de comprovação de fraude escolar, os pesos de todos os envolvidos serão automaticamente anulados.
- A documentação explicativa do projeto (**README**) é obrigatória .

Normas de confecção do manual do usuário (README)

- **COMO INICIAR/TERMINAR:** descrevendo como iniciar e terminar a execução do sistema estando na tela principal.
- **OPÇÕES OFERECIDAS:** descrevendo todas as opções oferecidas pelo sistema ao usuário.
- **PRINCIPAIS TELAS:** apresentando as principais telas de interface com o usuário.
- **CONCLUSÃO:** apresentando considerações finais sobre o sistema, limitações, problemas, etc.