

# Trabalho Final

---

- Esse trabalho deve ser realizado em grupos de 3 integrantes, salvo exceções justificadas e aceitas pela professora.
  - Apenas um dos integrantes do grupo deve enviar os arquivos do trabalho no Classroom, porém todos os integrantes devem marcar a atividade como concluída para que seja atribuída a nota.
  - Não se esqueçam de colocar o nome dos integrantes no início do arquivo que tem a função main. Ex:  
# Esse trabalho foi feito pelos integrantes:  
# Narizinho  
# Pedrinho  
# Visconde de Sabugosa
  - O código deve estar organizado, limpo e **MODULARIZADO**. De preferência as funções em um arquivo e a main em outro.
  - Documentação no código é fundamental
  - O código deve ser enviado até 06/06. Atrasos serão penalizados.
  - Cópias serão zeradas
  - A pontuação do trabalho e da prova oral estão interligadas, logo o trabalho deve estar funcionando até a data da prova oral, se o aluno não souber explicar o trabalho, a pontuação relativa ao trabalho será passível de anulação.
  - É proibido usar bibliotecas e funções que não foram vistas no curso. • **CUIDADO AO USAR `int(4)`, `str('oi')`, `float(9.0)`, ..., expressões desse tipo serão severamente penalizadas**
- 

## Batalha naval

O objetivo do trabalho é a implementação do jogo de batalha naval. O código deve conter as seguintes especificações:

- O jogo deve ter as seguintes opções: sair, recomeçar o jogo, e ver quantos e quais navios ainda não foram abatidos e quantos e quais já foram abatidos. Em qualquer momento do jogo os usuários devem ser capazes de realizar essas ações.
- No início do jogo deve-se criar automaticamente os dois tabuleiros, onde a criação

do tabuleiro não deve ser conhecida previamente pelo programador. Fica opcional ter outra possibilidade de inicialização do tabuleiro, onde os jogadores criam seus próprios tabuleiros.

- O tabuleiro tem 10 linhas e 10 colunas, o tabuleiro deve mostrar ~ quando a posição ainda não tiver tido nenhum tiro(água), \* quando houve um tiro, mas não acertou nenhuma embarcação e X quando alguma parte da embarcação for atingida. Como na figura abaixo:

A	B	C	D	E	F	G	H	I	J	
~	~	~	~	~	~	~	~	~	~	0
~	~	~	~	~	~	~	~	~	~	1
~	~	~	~	~	~	~	~	~	~	2
~	~	~	~	~	~	~	~	~	~	3
~	~	~	~	~	~	~	~	~	~	4
~	~	~	~	~	~	~	~	~	~	5
~	~	~	~	~	~	~	~	~	~	6
~	~	~	~	~	~	~	~	~	~	7
~	~	~	~	~	~	~	~	~	~	8
~	~	~	~	~	~	~	~	~	~	9

• jogador 1

A	B	C	D	E	F	G	H	I	J	
~	~	~	~	~	~	~	~	~	~	0
~	~	~	~	~	~	~	~	~	~	1
~	~	~	~	~	~	~	~	~	~	2
~	~	~	~	~	~	~	~	~	~	3
~	~	~	~	~	~	~	~	~	~	4
~	~	~	~	~	~	~	~	~	~	5
~	~	~	~	~	~	~	~	~	~	6
~	~	~	~	~	~	~	~	~	~	7
~	~	~	~	~	~	~	~	~	~	8
~	~	~	~	~	~	~	~	~	~	9

jogador 2

- O jogo deve solicitar ao usuário as posições no tabuleiro a serem acertadas, se o usuário acertar parte da embarcação o jogo deve imprimir o tabuleiro com a nova posição atingida, além de solicitar uma nova jogada para o mesmo jogador. Quando a embarcação é totalmente atingida, o jogo deve mostrar qual tipo de embarcação foi abatida.
- Navios: 1 porta-aviões (1x5 ou 5x1); 2 navios-tanque (1x4 ou 4x1); 3 contratorpedeiros (1x3 ou 3x1) ; 4 submarinos (1x2 ou 2x1). Os navios não podem ficar sobrepostos.
- O jogo não deve aceitar posições já receberam tiros.