

# Jogo da velha

em Portugol



# código principal

O código principal faz somente  
a chamada de procedimentos

```
Inicio  
ji()  
repita  
    contador()  
    jogada()  
fim()  
ate verific = verdadeiro  
Fimalgoritmo
```

# PROCEDIMENTO JI()

Este procedimento inicializa o tabuleiro com os números de 1 a 9 para representar as posições iniciais do jogo.

```
procedimento ji()  
inicio  
game[0,0]<-("1")  
game[0,1]<-("2")  
game[0,2]<-("3")  
game[1,0]<-("4")  
game[1,1]<-("5")  
game[1,2]<-("6")  
game[2,0]<-("7")  
game[2,1]<-("8")  
game[2,2]<-("9")  
fimprocedimento
```

# PROCEDIMENTO JOGO()

Este procedimento é responsável por exibir o tabuleiro do jogo da velha.

```
=====
JOGO DA VELHA
=====
+--+--+--+
| 1 | 2 | 3 |
+--+--+--+
| 4 | 5 | 6 |
+--+--+--+
| 7 | 8 | 9 |
+--+--+--+
```

```
procedimento jogo()
inicio
  escreval("=====")
  escreval("JOGO DA VELHA")
  escreval("=====")
  escreval("+--+--+--+")
  para i de 0 ate 2 faca
    escreval("|", game[i,0], " |", game[i,1], " |", game[i,2], " |")
    escreval("+--+--+--+")
  fimpara
fimprocedimento
```

# PROCEDIMENTO CONTADOR()

Este procedimento decide qual jogador está na vez (X ou O) com base no número total de jogadas feitas (Count). Cada jogador tem uma jogada a mais que o outro, começando com X. Se 'Count' for igual a um número par, a jogada é da vez de X, senão a vez é de O.

```
procedimento contador()
inicio
se (count mod 2) = 0 entao
    player<-"X"
senao
    Player<-"O"
fimse
count<-count+1
fimprocedimento
```

# PROCEDIMENTO JOGADA()

- Este procedimento é responsável por receber a jogada do jogador atual. Ele pede ao jogador para informar a posição desejada.
- O código valida a entrada para garantir que seja um número entre 1 e 9 e que a posição escolhida não esteja ocupada. Caso contrário, pede que o jogador escolha outra posição.
- A jogada é então registrada no tabuleiro.

# PROCEDIMENTO FIM()

Este procedimento verifica se houve um vencedor ou se o jogo terminou em empate.

Ele verifica todas as possíveis combinações de vitória no tabuleiro e exibe a mensagem de vitória se alguma delas for alcançada.

Se o tabuleiro estiver completo e não houver vencedor, é declarado empate.

No final, pergunta se os jogadores desejam jogar novamente.

```
=====
JOGO DA VELHA
=====
+---+---+---+
|X |O |3 |
+---+---+---+
|X |O |6 |
+---+---+---+
|X |8 |9 |
+---+---+---+

Vitória de: X
deseja jogar de novo?
s para sim / n para não
```

```
=====
JOGO DA VELHA
=====
+---+---+---+
|X |O |X |
+---+---+---+
|4 |O |X |
+---+---+---+
|7 |O |9 |
+---+---+---+

Vitória de: O
deseja jogar de novo?
s para sim / n para não
|
```

```
=====
JOGO DA VELHA
=====
+---+---+---+
|X |X |O |
+---+---+---+
|O |X |X |
+---+---+---+
|X |O |O |
+---+---+---+

Ninguem ganhou
deseja jogar de novo?
s para sim / n para não
```

# PROCEDIMENTO procfim()

- Válida a entrada do jogador para a pergunta "deseja jogar de novo?"
- Caso o jogador deseje jogar novamente este procedimento reinicia o jogo, caso contrário ele é responsável por permitir que o jogo finalize

```
procedimento procfim()
var
procverif : logico
nj : caractere
inicio
repita
leia(nj)
se ((nj= "s")ou(nj= "S")) entao
    count<-0
    ji()
    procverif <- verdadeiro
senao
    se ((nj = "n") ou (nj = "N")) entao
        verif<-verdadeiro
        procverif <- verdadeiro
    senao
        escreval("opção inválida")
    fimse
fimse
ate procverif = verdadeiro
fimprocedimento
```



# VÁRIÁVEIS GLOBAIS

- game : é a matriz que armazena o tabuleiro do jogo
- i : variável do tipo inteiro usada para controlar posição das linhas
- count : variável do tipo inteiro usada para alternar entre os jogadores
- player : variável do tipo caractere usada para armazenar os jogadores "X" e "O"
- verif : variável do tipo lógico que mantém o jogo em execução até que jogador deseje finalizar

# VÁRIÁVEIS LOCAIS

## **jogada**

movimento: variável do tipo inteiro que recebe a posição de 1 a 9 onde o jogador deseja jogar  
j: variável do tipo inteiro usada para definir em que posição o jogador deseja realizar um movimento  
valid: variável lógica que mantém o procedimento em execução até o jogador inserir uma jogada válida

## **fim**

novoj: variável lógica que valida se houve vitória ou empate

## **procfim**

procverif : variável lógica que mantém o procedimento em execução até que a entrada seja validada  
nj: variável do tipo caractere que recebe resposta da pergunta