

## Referência da Classe Tabuleiro

---

Classe que representa o tabuleiro de um jogo. [Mais...](#)

```
#include <Tabuleiro.hpp>
```

## Membros Públicos

---

**Tabuleiro** ()

Construtor padrão da classe **Tabuleiro**.

void **configurarTabuleiro** (int linha, int col)

Configura o tamanho do tabuleiro com o número de linhas e colunas especificadas.

void **definirPosicao** (int \_x, int \_y, char \_peca)

char **obterPeca** (int \_x, int \_y)

Obtém o conteúdo da célula na posição especificada do tabuleiro.

bool **posicaoValida** (int \_x, int \_y)

Verifica se a posição especificada está dentro dos limites do tabuleiro.

void **imprimir** ()

Imprime o estado atual do tabuleiro.

**~Tabuleiro** ()

Destrutor da classe **Tabuleiro**.

---

## Descrição detalhada

---

Classe que representa o tabuleiro de um jogo.

A classe **Tabuleiro** gerencia a estrutura do tabuleiro, incluindo o número de linhas e colunas, e fornece métodos para inicializar e definir posições no tabuleiro. Ela também possui um método para imprimir o estado atual do tabuleiro.

## Construtores e Destrutores

---

◆ **Tabuleiro**()

Tabuleiro::Tabuleiro ( )

Construtor padrão da classe **Tabuleiro**.

Inicializa o tabuleiro com dimensões padrão.

```
4      : linhas(0), colunas(0), matrizTabuleiro(nullptr) {  
5      // Inicializa o número de linhas e colunas e a matrizTabuleiro  
6      }
```

◆ ~Tabuleiro()

Tabuleiro::~~Tabuleiro ( )

Destrutor da classe **Tabuleiro**.

Libera qualquer recurso alocado pela classe **Tabuleiro**.

```
48      {  
49      if (matrizTabuleiro) {  
50          for (int i = 0; i < linhas; i++) {  
51              delete[] matrizTabuleiro[i];  
52          }  
53          delete[] matrizTabuleiro;  
54      }  
55      }
```

## Documentação das funções

◆ configurarTabuleiro()

```
void Tabuleiro::configurarTabuleiro ( int linha,  
                                     int col )
```

Configura o tamanho do tabuleiro com o número de linhas e colunas especificadas.

### Parâmetros

**linha** Número de linhas do tabuleiro.

**col** Número de colunas do tabuleiro.

Configura o tabuleiro para ter as dimensões especificadas.

```
8      {  
9      linhas = _linhas;  
10     colunas = _colunas;  
11  
12     // Aloca matriz para o tabuleiro  
13     matrizTabuleiro = new char*[linhas];  
14     for (int i = 0; i < linhas; i++) {  
15         matrizTabuleiro[i] = new char[colunas];  
16         for (int j = 0; j < colunas; j++) {  
17             matrizTabuleiro[i][j] = '.'; // inicializa as posições com ponto  
18         }  
19     }  
20 }
```

### ◆ definirPosicao()

```
void Tabuleiro::definirPosicao ( int  _x,  
                                int  _y,  
                                char _peca )
```

```
22      {  
23      if (posicaoValida(_x, _y)) {  
24          matrizTabuleiro[_x][_y] = _peca;  
25      }  
26 }
```

### ◆ imprimir()

```
void Tabuleiro::imprimir ( )
```

Imprime o estado atual do tabuleiro.

Imprime o tabuleiro no estado atual.

```
39     {
40         for (int i = 0; i < linhas; i++) {
41             for (int j = 0; j < colunas; j++) {
42                 std::cout << matrizTabuleiro[i][j] << " ";
43             }
44             std::cout << std::endl;
45         }
46     }
```

### ♦ obterPeca()

```
char Tabuleiro::obterPeca ( int _x,
                           int _y )
```

Obtém o conteúdo da célula na posição especificada do tabuleiro.

#### Parâmetros

**\_x** Coordenada x da posição no tabuleiro.

**\_y** Coordenada y da posição no tabuleiro.

#### Retorna

Caracter que representa o conteúdo da célula na posição especificada.

```
28     {
29         if (posicaoValida(_x, _y)) {
30             return matrizTabuleiro[_x][_y];
31         }
32         return '.'; // retorna um '.' se a posição for inválida
33     }
```

### ♦ posicaoValida()

```
bool Tabuleiro::posicaoValida ( int _x,  
                               int _y )
```

Verifica se a posição especificada está dentro dos limites do tabuleiro.

### Parâmetros

**\_x** Coordenada x da posição no tabuleiro.

**\_y** Coordenada y da posição no tabuleiro.

### Retorna

Verdadeiro se a posição estiver dentro dos limites, falso caso contrário.

```
35  
36     return (_x >= 0 && _x < linhas && _y >= 0 && _y < colunas);  
37 }
```

A documentação para essa classe foi gerada a partir dos seguintes arquivos:

- /Users/iangodoi/Desktop/TP-jogosTabuleiro-cpp-desenvolvimento/include/**Tabuleiro.hpp**
- /Users/iangodoi/Desktop/TP-jogosTabuleiro-cpp-desenvolvimento/src/**Tabuleiro.cpp**