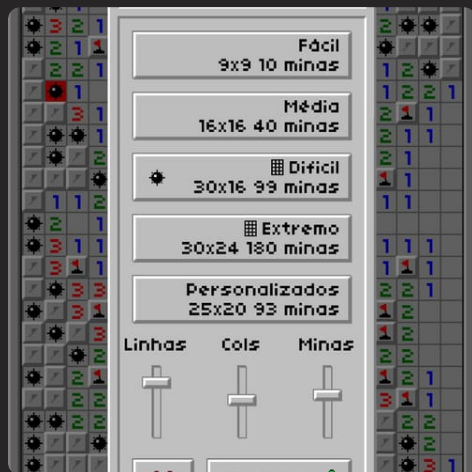




# Campo minado em Python e Java

Você sabe jogar campo minado? Aprenda a construir esse clássico jogo com Python e Java neste trabalho de campo minado!

# Introdução ao Campo Minado



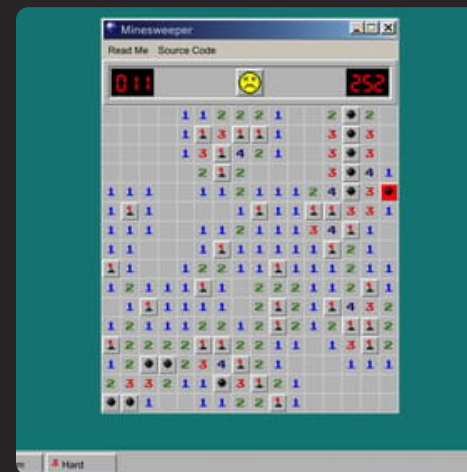
## O que é Campo Minado?

O Campo Minado é um jogo eletrônico inventado pela Microsoft em 1989. O objetivo é encontrar todas as minas que estão escondidas no tabuleiro.



## Como funciona o Jogo?

O jogo é composto por um tabuleiro retangular, algumas casas possuem bombas, outras possuem um número, que indica quantas bombas aquela casa toca.



## O que são os números?

Os números são dicas que ajudam a encontrar e sinalizar as bombas. Por exemplo: Se uma casa tiver o número 3, significa que há 3 bombas adjacentes a ela.

# Regras do Jogo

## Dicas

Não clique em casas aleatórias para evitar que uma bomba exploda. Marque as casas com bombas com a bandeira, para indicar que ali é um local perigoso. Use a lógica e a matemática para encontrar as bombas.

1

### Como jogar?

O objetivo do jogo é encontrar todas as bombas escondidas no tabuleiro. Cada casa revelada ajuda a encontrar as bombas, já que os números mostram a quantidade de bombas adjacentes.

2

3

### Objetivo

Ganha quem encontrar todas as minas sem explodir nenhuma casa. É um jogo de desafio intelectual e paciência!

# Estratégias para Jogar

## **Analise os números**

Quando houver um número no tabuleiro, você pode ter uma pista de que existem bombas nas casas ao redor. Use isso para encontrar as bombas.

## **Isolar as Bombas**

Você pode isolar as bombas para evitar acidentes. Use a bandeira para marcar as casas com bombas e assim isolá-las, assim você não esquece quais são as casas perigosas.

## **Use a Lógica Matemática**

Usando a lógica você pode chegar na solução. Se você souber que uma casa tem dois vizinhos que têm uma mina, você pode deduzir que a terceira casa não tem uma mina.

## **Siga um Plano**

Se você está tendo dificuldades para encontrar as bombas, siga um plano para marcar as casas que sua lógica eliminou. Se você usa um padrão para marcar as casas, você pode focar as energias nas casas incertas.

# Código do jogo em Python:

```
import random

class CampoMinado:
    def __init__(self):
        self.linhas = 8
        self.colunas = 8
        self.minas = random.randint(5, 10)
        self.tabuleiro = [[0] * self.colunas for _ in range(self.linhas)]
        self.cobertura = ["-"] * self.colunas for _ in range(self.linhas)]
        self.jogadas_restantes = self.linhas * self.colunas - self.minas
        self.gerar_minas()

    def gerar_minas(self):
        minas_plantadas = 0
        while minas_plantadas < self.minas:
            x = random.randint(0, self.linhas - 1)
            y = random.randint(0, self.colunas - 1)
            if self.tabuleiro[x][y] != -1:
                self.tabuleiro[x][y] = -1
                minas_plantadas += 1
                self.atualizar_numeros(x, y)

    def atualizar_numeros(self, x, y):
        for i in range(max(0, x - 1), min(self.linhas - 1, x + 1) + 1):
            for j in range(max(0, y - 1), min(self.colunas - 1, y + 1) + 1):
                if self.tabuleiro[i][j] != -1:
                    self.tabuleiro[i][j] += 1
```

```

def jogar(self, x, y):
    if self.tabuleiro[x][y] == -1:
        return False
    elif self.tabuleiro[x][y] == 0:
        self.expandir_zeros(x, y)
    else:
        self.cobertura[x][y] = str(self.tabuleiro[x][y])
    self.jogadas_restantes -= 1
    return True

def expandir_zeros(self, x, y):
    fila = [(x, y)]
    visitados = set([(x, y)])
    while fila:
        i, j = fila.pop(0)
        self.cobertura[i][j] = "0"
        for a in range(max(0, i - 1), min(self.linhas - 1, i + 1) + 1):
            for b in range(max(0, j - 1), min(self.colunas - 1, j + 1) + 1):
                if self.tabuleiro[a][b] == 0 and (a, b) not in visitados:
                    fila.append((a, b))
                    visitados.add((a, b))
                elif self.tabuleiro[a][b] != -1:
                    self.cobertura[a][b] = str(self.tabuleiro[a][b])
    self.jogadas_restantes -= len(visitados) - 1

def imprimir_tabuleiro(self):
    print(" ", end=" ")
    for coluna in range(self.colunas):
        print(coluna, end=" ")
    print()
    for i, linha in enumerate(self.cobertura):
        print(i, end=" ")
        print(" ".join(linha))

```

```
def imprimir_tabuleiro_com_minas(self):
    print(" ", end=" ")
    for coluna in range(self.colunas):
        print(coluna, end=" ")
    print()
    for i in range(self.linhas):
        print(i, end=" ")
        for j in range(self.colunas):
            if self.tabuleiro[i][j] == -1:
                print("*", end=" ")
            else:
                print(self.tabuleiro[i][j], end=" ")
        print()
```

*# Exemplo de uso*

```
jogo = CampoMinado()
jogo.imprimir_tabuleiro()
```

```
while True:
    x = int(input("Digite o número da linha: "))
    y = int(input("Digite o número da coluna: "))
    if not jogo.jogar(x, y):
        print("Você perdeu!")
        print("Tabuleiro com as minas:")
        jogo.imprimir_tabuleiro_com_minas()
        break
    jogo.imprimir_tabuleiro()
    if jogo.jogadas_restantes == 0:
        print("Você ganhou!")
        break
```

# Código do jogo em Java:

```
public class CampoMinado {
    private int linhas;
    private int colunas;
    private int minas;
    private int[][] tabuleiro;
    private String[][] cobertura;
    private int jogadas_restantes;

    public CampoMinado() {
        this.linhas = 8;
        this.colunas = 8;
        this.minas = new Random().nextInt(6) + 5;
        this.tabuleiro = new int[this.linhas][this.colunas];
        this.cobertura = new String[this.linhas][this.colunas];
        this.jogadas_restantes = this.linhas * this.colunas - this.minas;
        gerarMinas();
        inicializarCobertura();
    }

    private void gerarMinas() {
        int minas_plantadas = 0;
        Random random = new Random();
        while (minas_plantadas < this.minas) {
            int x = random.nextInt(this.linhas);
            int y = random.nextInt(this.colunas);
            if (this.tabuleiro[x][y] != -1) {
                this.tabuleiro[x][y] = -1;
                minas_plantadas++;
                atualizarNumeros(x, y);
            }
        }
    }
}
```



```
private void atualizarNumeros(int x, int y) {  
    for (int i = Math.max(0, x - 1); i <= Math.min(this.linhas - 1, x + 1); i++) {  
        for (int j = Math.max(0, y - 1); j <= Math.min(this.colunas - 1, y + 1); j++) {  
            if (this.tabuleiro[i][j] != -1) {  
                this.tabuleiro[i][j]++;  
            }  
        }  
    }  
}  
  
public boolean jogar(int x, int y) {  
    if (this.tabuleiro[x][y] == -1) {  
        return false;  
    } else if (this.tabuleiro[x][y] == 0) {  
        expandirZeros(x, y);  
    } else {  
        this.cobertura[x][y] = String.valueOf(this.tabuleiro[x][y]);  
    }  
    this.jogadas_restantes--;  
    return true;  
}
```

```

private void expandirZeros(int x, int y) {
    List<int[]> fila = new ArrayList<>();
    fila.add(new int[]{x, y});
    boolean[][] visitados = new boolean[this.linhas][this.colunas];
    visitados[x][y] = true;

    while (!fila.isEmpty()) {
        int[] posicao = fila.remove(0);
        int i = posicao[0];
        int j = posicao[1];
        this.cobertura[i][j] = "0";

        for (int a = Math.max(0, i - 1); a <= Math.min(this.linhas - 1, i + 1); a++) {
            for (int b = Math.max(0, j - 1); b <= Math.min(this.colunas - 1, j + 1); b++) {
                if (this.tabuleiro[a][b] == 0 && !visitados[a][b]) {
                    fila.add(new int[]{a, b});
                    visitados[a][b] = true;
                } else if (this.tabuleiro[a][b] != -1) {
                    this.cobertura[a][b] = String.valueOf(this.tabuleiro[a][b]);
                }
            }
        }
    }
    this.jogadas_restantes -= getNumeroVisitados(visitados) - 1;
}

```

```
private int getNumeroVisitados(boolean[][] visitados) {  
    int count = 0;  
    for (boolean[] linha : visitados) {  
        for (boolean visitado : linha) {  
            if (visitado) {  
                count++;  
            }  
        }  
    }  
    return count;  
}  
  
public void imprimirTabuleiro() {  
    System.out.print(" ");  
    for (int coluna = 0; coluna < this.colunas; coluna++) {  
        System.out.print(coluna + " ");  
    }  
    System.out.println();  
    for (int i = 0; i < this.linhas; i++) {  
        System.out.print(i + " ");  
        for (int j = 0; j < this.colunas; j++) {  
            System.out.print(this.cobertura[i][j] + " ");  
        }  
        System.out.println();  
    }  
}
```

```
public void imprimirTabuleiroComMinas() {  
    System.out.print(" ");  
    for (int coluna = 0; coluna < this.colunas; coluna++) {  
        System.out.print(coluna + " ");  
    }  
    System.out.println();  
    for (int i = 0; i < this.linhas; i++) {  
        System.out.print(i + " ");  
        for (int j = 0; j < this.colunas; j++) {  
            if (this.tabuleiro[i][j] == -1) {  
                System.out.print("* ");  
            } else {  
                System.out.print(this.tabuleiro[i][j] + " ");  
            }  
        }  
        System.out.println();  
    }  
}  
  
private void inicializarCobertura() {  
    for (int i = 0; i < this.linhas; i++) {  
        for (int j = 0; j < this.colunas; j++) {  
            this.cobertura[i][j] = "-";  
        }  
    }  
}
```

```
public static void main(String[] args) {  
    CampoMinado jogo = new CampoMinado();  
    jogo.imprimirTabuleiro();  
  
    Scanner scanner = new Scanner(System.in);  
    while (true) {  
        System.out.print("Digite o número da linha: ");  
        int x = scanner.nextInt();  
        System.out.print("Digite o número da coluna: ");  
        int y = scanner.nextInt();  
        if (!jogo.jogar(x, y)) {  
            System.out.println("Você perdeu!");  
            System.out.println("Tabuleiro com as minas:");  
            jogo.imprimirTabuleiroComMinas();  
            break;  
        }  
        jogo.imprimirTabuleiro();  
        if (jogo.jogadas_restantes == 0) {  
            System.out.println("Você ganhou!");  
            break;  
        }  
    }  
}
```



**FIM.**