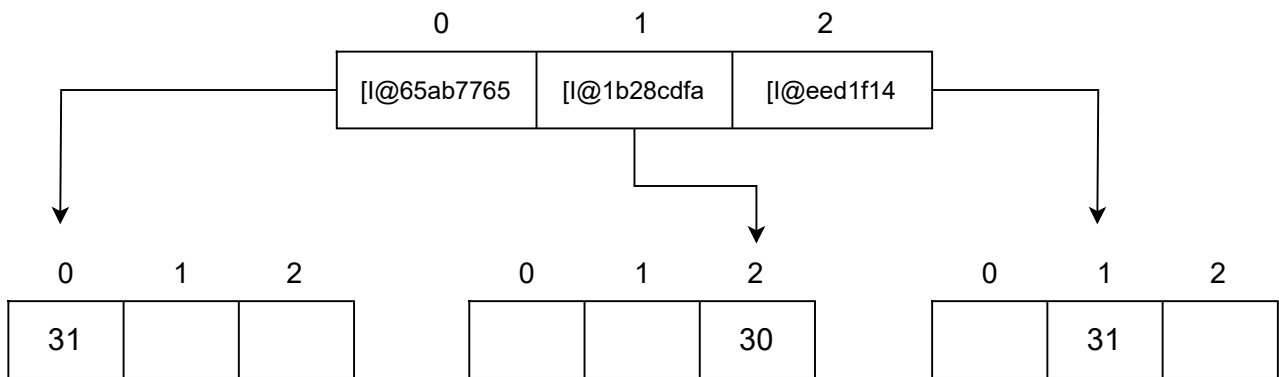


```
int[][] dias = new int[3][3];
```

## Representação 1

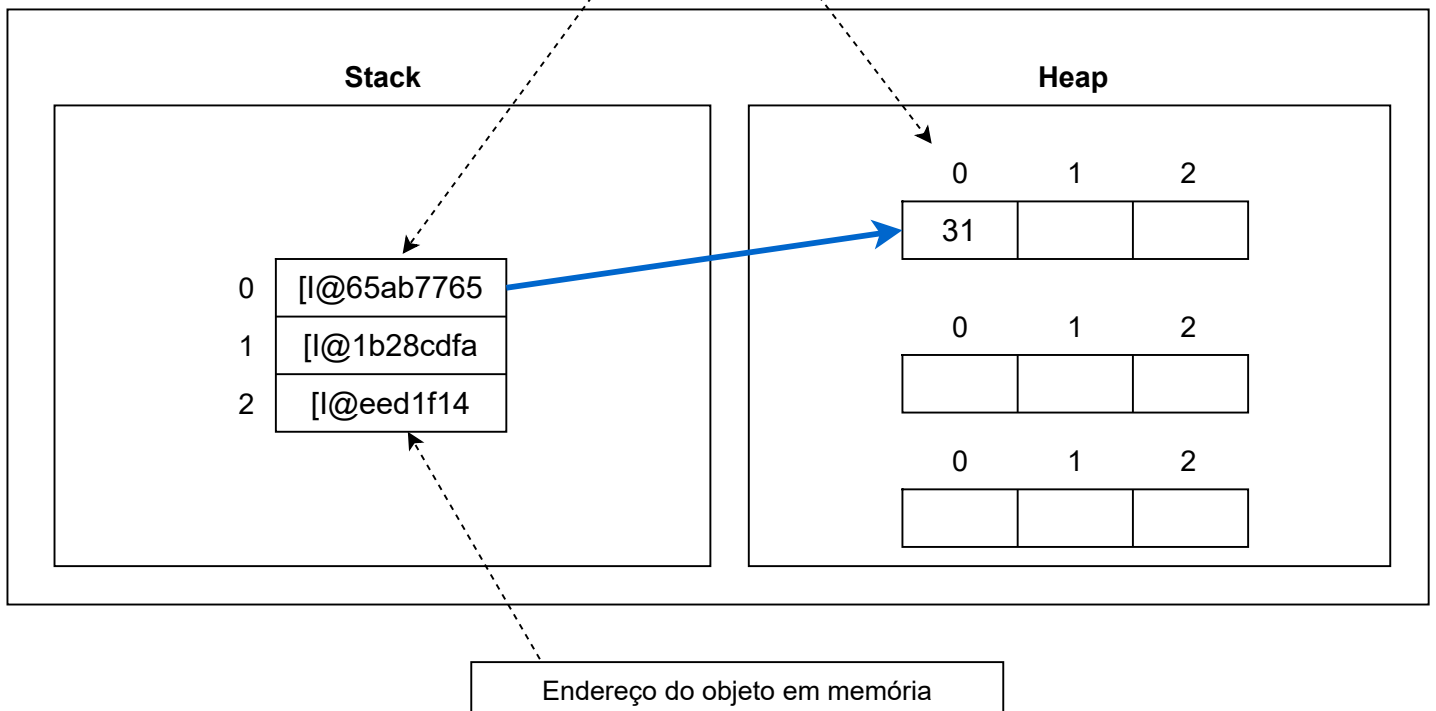
```
dias[0][0] = 31;      dias[1][2] = 30;      dias[2][1] = 31;
```



## Representação 2

```
dias[0][0] = 31;
```

### Representação em memória



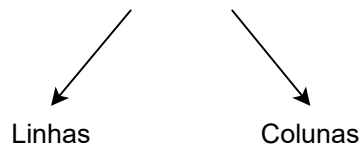
## Representação 3 - a mais conhecida e fácil de entender

---

`dias[0][0] = 31;`

`dias[1][2] = 28;`

`dias[2][1] = 25;`



		Colunas		
		0	1	2
Linhas	0	31		
	1			28
	2		25	

### Loop aninhado

A estrutura abaixo se chama loop aninhado, ou seja, é o uso de uma estrutura dentro de outra para acessar elementos, normalmente dentro de arrays multidimensionais.

Neste exemplo o for mais externo é responsável por percorrer as linhas da matriz, enquanto o for mais interno é responsável por percorrer as colunas. Enquanto uma linha não é totalmente percorrida, os elementos da próxima linha não serão acessados.

A instrução `dias[linha][coluna]` imprime o resultado de cada iteração: o primeiro colchete informa a linha que está sendo iterada; o segundo os elementos dessa linha.

```
for (int linha = 0; linha < dias.length; linha++)
{
    for (int coluna = 0; coluna < dias[linha].length; coluna++)
    {
        System.out.println( dias[linha][coluna] );
    }
}
```