Estruturas de Repetição

FUNDAMENTOS DE PROGRAMAÇÃO

Laços de Repetição

APRESENTAÇÃO DO CONTEÚDO

O que iremos ver:

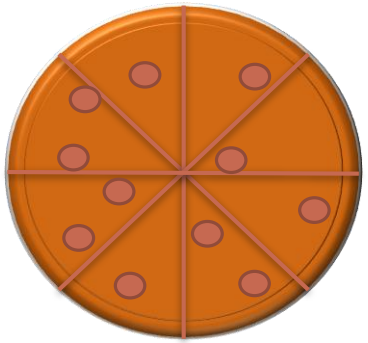
- Repetições com teste lógico no início
- Repetições com teste lógico no final
- Repetições com variável de controle
- Execícios propostos

INTRODUÇÃO

Falaremos sobre repetições, que podem ser chamadas de laços.

Imaginemos uma pizza como exemplo:

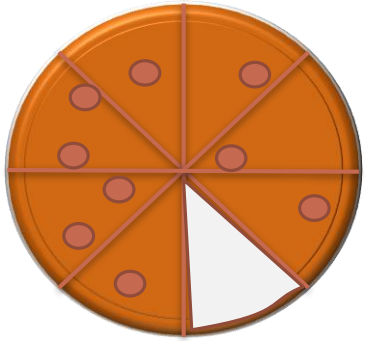
- Temos uma função(método) `comerFatia();`



INTRODUÇÃO(EXEMPLO)

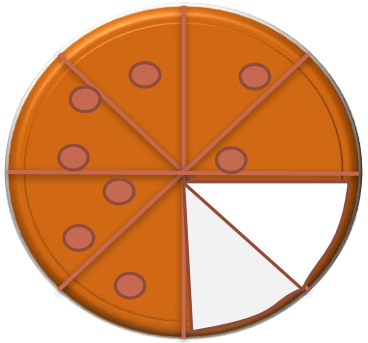
Tendo o método comer fatia, vamos executar ela de forma sequencial, que é como normalmente é feita a execução de um programa, observando com mais atenção o exemplo do programa, como também a demostração mais para frente dos slides, notamos que essa tarefa comer fatia é repetida até que a pizza acabe, ou seja, enquanto não acaba a pizza, vamos comer uma fatia.

```
comerFatia();
comerFatia();
comerFatia();
    ...
comerFatia();
```



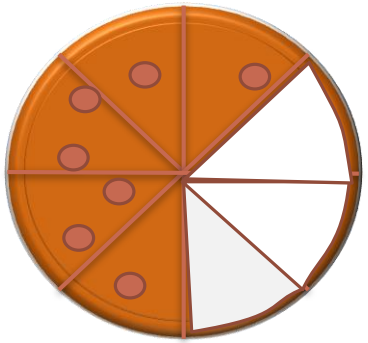
INTRODUÇÃO(EXEMPLO)

```
comerFatia();  // Tem fatia? Ohh, tem. Então segue se saboriando. Ou seja true.
```



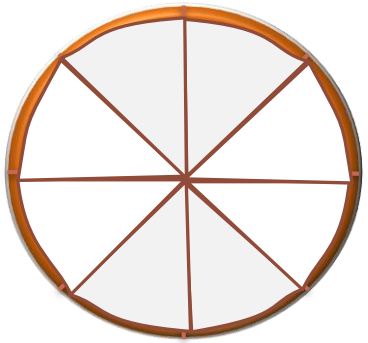
INTRODUÇÃO(EXEMPLO)

```
comerFatia();  // Tem fatia? Ohh, tem. Então segue
```
se saboriando, e vai até acabar ...



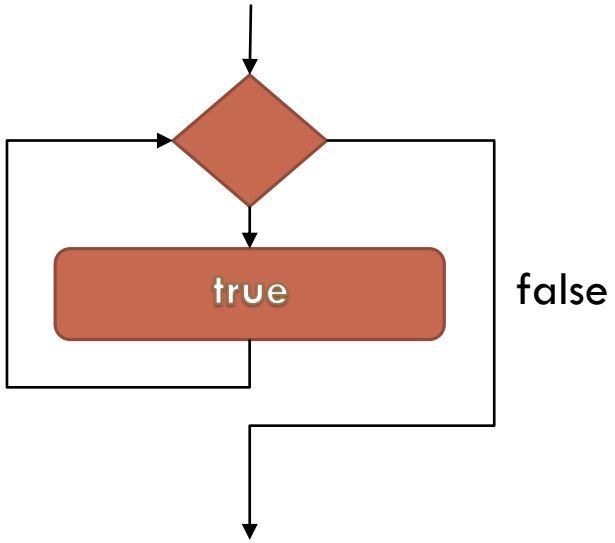
INTRODUÇÃO(EXEMPLO)

```
comerFatia();
```
// Ohh, espere um pouco, não há mais fatias, sendo assim saímos da repetição. Ou seja quando a condição é false saímos do laço repetitivo.



REPETIÇÃO COM TESTE LÓGICO NO ÍNICIO

Tal como as condições, as repetições fazem um teste lógico(verdadeiro ou falso), se for verdadeiro faz um bloco e retorna a condição, e chamamos de loop(repetição) e faz o teste enquanto a condição for verdade, caso contrário(falso) termina a repetição.



REPETIÇÃO COM TESTE LÓGICO NO ÍNICIO

Estrutura do while em java:

```
while (temFatia()) {
    comerFatia();
}
```

Vai a condição(true ou false)

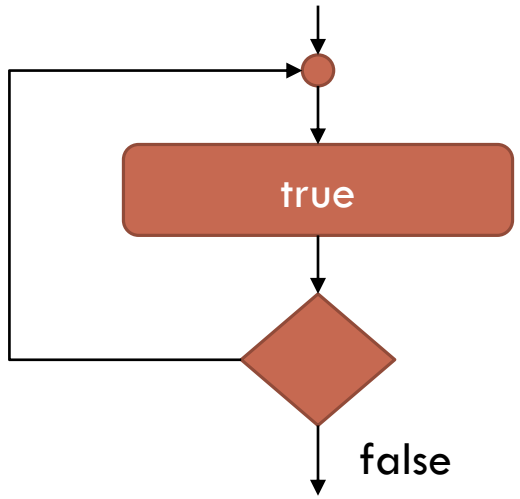
Vai a instrução a ser repetida

Vamos a um exemplo:

```
int c = 1;
while (c <= 6) {
    System.out.println(c);
    c++;
}
```

REPETIÇÃO COM TESTE LÓGICO NO FIM

Diferente da repetição anterior que é feito o teste lógico e depois executado o bloco, nesse laço é executado o bloco depois que é feito o teste lógico, ou seja, feito no final.



REPETIÇÃO COM TESTE LÓGICO NO FIM

Estrutura do do .. while em java:

```
do {
    comerFatia();
} while (temFatia());
```

Vai a instrução a ser repetida

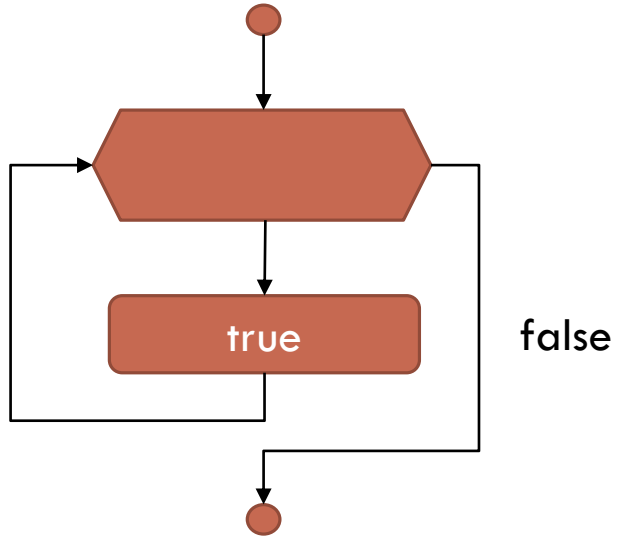
Vai a condição(true ou false)

Vamos a um exemplo:

```
int c = 1;
do {
    System.out.println(c);
    c++;
} while (c <= 6);
```

REPETIÇÃO COM VARIÁVEL DE CONTROLE

As estruturas de repetição com varáveis possuém três sessões importantes, ***(início; teste; incremento)*** e usamos a palavra reservada for para criar o laço. Vamos ter exemplos mais para frente, mas vamos entender em fluxograma antes:



REPETIÇÃO COM VARIÁVEL DE CONTROLE

Estrutura do for em java:

```
for (inicio; teste; incremento) {
    <instrução a repetir>;
}
```

Vamos a um exemplo:

```
for (int c=1; c <= 6; c++) {
    System.out.println(c);
}
```

PALAVRAS RESERVADAS(**BREAK E CONTINUE**)

Algumas linguagens de programação especificam ainda uma palavra reservada para sair da estrutura de repetição de dentro do bloco de código, "quebrando" a estrutura. No java não é excessão, usamos a palavra reservada:

break;

Também é oferecido por algumas linguagens uma palavra reservada para terminar uma iteração específica do bloco de código, forçando nova verificação da condição, ou seja, ignora uma iteração e vai para próxima.

continue;

FIM

Feito por:

- Anselmo Nhamage