Recursividade

Recursividade

Uma função **recursiva** é uma função que chama ela própria.

A recursividade produz repetição sem usar os comandos *while*, *do..while*, *for*.

Exemplo: A função fatorial é definida matematicamente por

$$n! = \begin{cases} 1, & \text{se } n = 0 \\ n * (n-1)!, & \text{se } n>0 \end{cases}$$

Esta é uma definição recursiva porque o fatorial aparece no lado direito, ou seja, a função é definida em termos dela mesma.

Recursividade

Função fatorial:

```
public int fat (int n) {
    if (n==0)
      return 1;
    else
      return n * fat (n-1);
}
```

Recursividade

Uma função recursiva deve ter uma parte base e uma parte recursiva.

A parte base é a que pára a recursão.

A parte recursiva é onde a função chama ela própria.

```
public int fat (int n) {
    if (n==0)
      return 1;
    else
      return n * fat (n-1);
}
```

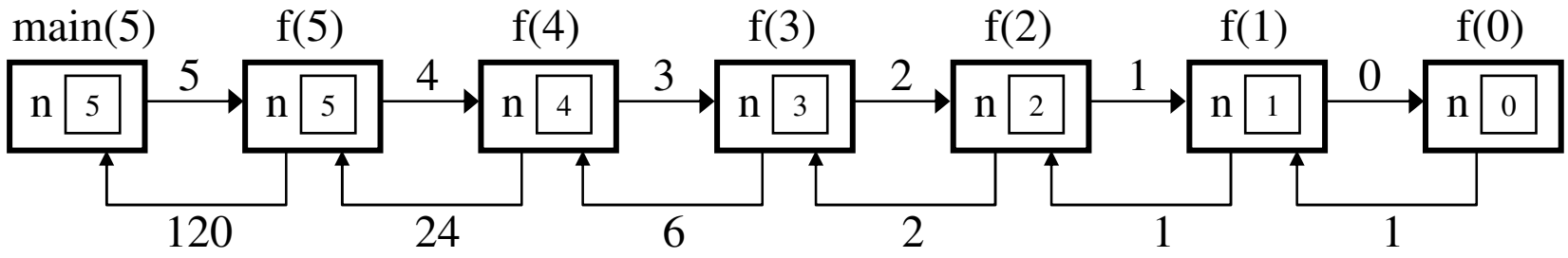
parte base

parte recursiva

Rastreamento de uma Chamada Recursiva

Exemplo: Chamada da função recursiva fat (5).

```
public int fat (int n) {
    if (n==0)
       return 1;
    else
       return n * fat (n-1);
}
```



Exercício

Implemente e teste uma lista recursiva que apresenta os seguintes métodos:

public void insere (E elemento)

public void insere (E elemento, int posicao)

public int retornaPosicao (E elemento)

public E retorna (int posicao)

public E remove (E elemento)

public void remove (int posicao)

public int numeroElementos()

public boolean estaVazia()

Exercício

```
public class ListaRecursiva<E> {
        E elemento;
        ListaRecursiva<E> resto;
        int numElementos;

public ListaRecursiva(){
}

public void insere (E elemento){
    if (this.elemento == null) {
        this.elemento = elemento;
        this.resto = new ListaRecursiva<E>();
    }
    else
        this.resto.insere(elemento);
    this.numElementos++;
}
```

Exercício

```
public int retornaPosicao (E elemento){
    if (this.elemento == null)
        return -1;
    else
        if (this.elemento.equals(elemento))
            return 0;
        else {
            int pos = this.resto.retornaPosicao(elemento);
            if (pos == -1)
                return pos;
            else
                return pos+1;
        }
}
...
}
```